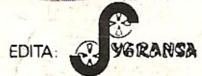


LOS MEJORES LISTADOS PARA TU MICRO

EDITA: 

·año·I·numero·I·

ADIVINAR LIST
por ordenador

TRUCOS

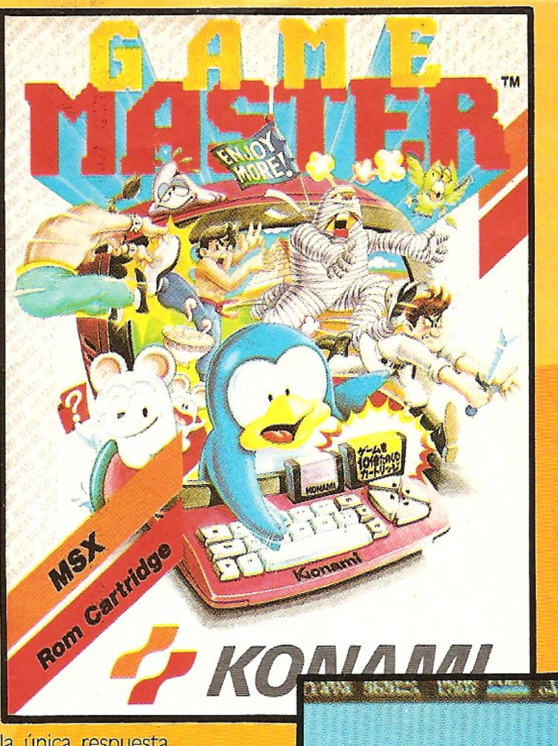
MGEN LIST
generador de sprites multicolor

la instruccion SOUND

mundo precio
PVP IVA INC 200 pesetas

KEOPS LIST

PVP CANARIAS 200 PTS

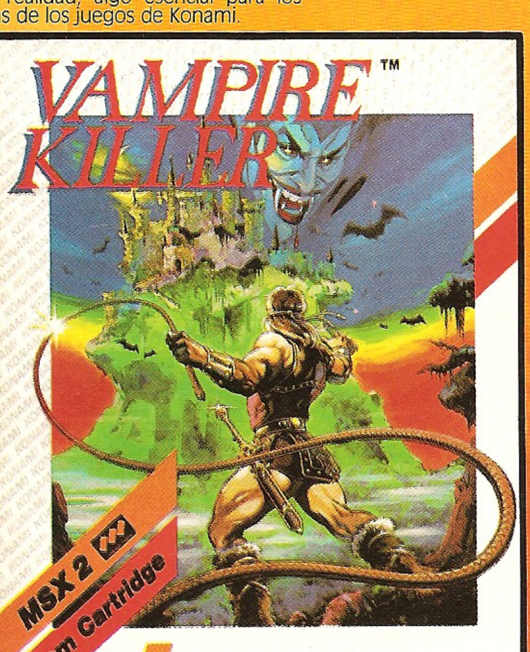


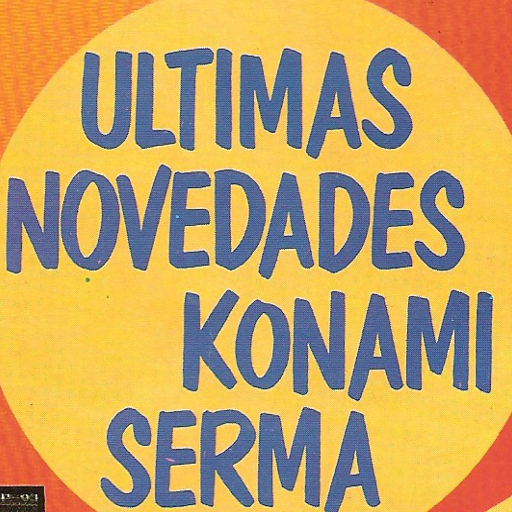
GAMESMASTER

Gamesmaster es la única respuesta para esos juegos difíciles.
Posibilidades de ralentizar el movimiento e incluso congelar el juego, modificar la velocidad y etapas del juego.
Volcados de pantalla por impresora, pasa a cinta o a disco las máximas puntuaciones. Selecciona el número de jugadores y calcula las máximas puntuaciones.
En realidad, algo esencial para los fans de los juegos de Konami.

PENGUIN ADVENTURE

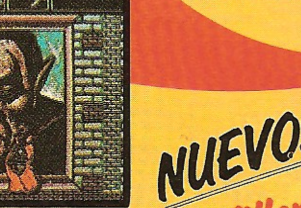
Guía a nuestro héroe Penguin por cuevas, mares y glaciares para devolver la normalidad a la Princesa Penguin y su reino.
Pelea con los Tiranosaurios y con montones de extraños enemigos utilizando los poderes comprados al Comerciante Esquimal.
Apuesta los peces en una máquina tragaperras para aumentar la puntuación y bucea por escenarios submarinos en un intento de restaurar el Paraíso Penguin.
Un juego lleno de acción con los gráficos que acostumbras esperar de Konami.









VAMPIRE KILLER

Enfréntate a Drácula en esta espectacular aventura de Konami. Atraviesa el camino del Diablo, consigue las armas y poderes especiales y quizá puedas atravesar el castillo satánico y luchar con el Maestro de la Oscuridad.
En cada esquina habrá adversarios que te helarán la sangre; necesitarás lograr los medios de superarlos. Una decisión equivocada te hará fracasar...
Este cartucho ROM con un Mega lleno de acción, con los brillantes gráficos a que Konami nos tiene acostumbrados, es una aventura del principio al final.

NUEVOS CARTUCHOS

Vampire Killer - 6.800 pts. (solo MSX 2)
Penguin Adventure - 6.150 pts.
Game Master - 6.150 pts.

ENVIA ESTE CUPON A KONAMI SHOP. FRANCISCO NAVACERRADA, 19. 28028 MADRID
TITULO:
NOMBRE Y APELLIDOS:
DIRECCION:
COD. POSTAL:
POBLACION:
PROVINCIA:
FORMA DE PAGO: CONTRARREMBOLSO ☐
POR TALON BANCARIO ☐

KONAMI ES DISTRIBUIDO EXCLUSIVAMENTE EN ESPAÑA POR SERMA: CARDENAL BELLUGA, 21. 28028 MADRID. TLFOS: 256 21 01 - 12 22.

EDITORIAL

Tranquilos, nuestro personaje de portada no está enfadado, ni mucho menos, simplemente quiere decir algo así como: «aquí estoy yo», y parece pero que muy dispuesto a mantenerse erguido contra viento y marea.
Quizás sea una casualidad, pero los que hacemos esta revista, no sabemos muy bien por qué, pero el caso es que nos resulta simpática la imagen descrita.

Por supuesto este personaje es de ficción, no es en absoluto real, sin embargo su significado no está exento de un trasfondo que sí tiene algo de real.

Es un poco lo que sucede con los juegos de ordenador, son irreales en tanto que nos descubren un mundo que no existe, pero comienzan a cobrar realidad en cuato nos dejamos subyugar por él y lo convertimos en escenario de una parte de nuestra vida.

Cada cual tendrá sus propias observaciones sobre los juegos, la anterior es una más. Lo que sí es cierto es que a una buena parte de la humanidad, los juegos nos interesan muchísimo, y si además son juegos de ordenador, aumenta nuestro interés. Y claro, si hemos optado por el sistema de ordenadores MSX, lo normal es que nos lo pasemos en grande.

Desde esta revista queremos hacer nuestro aporte a tan «noble causa» de pasarlo bien. Ya que el interés va a estar centrado en la programación con MSX, donde sin duda estará de protagonista el JUEGO, con mayúsculas, y su mundo.

Reivindicamos el juego, pero acompañándolo de otros aspectos no menos importantes de la programación, ya que quedarnos ahí sería infrautilizar nuestro ordenador.

Ayudar a que nuestros lectores saquen el máximo provecho de su ordenador, es en suma nuestro objetivo.

Y queremos hacerlo de una forma eminentemente práctica, útil y sencilla: listados, información y algo de teoría.

Esperamos que sea de vuestro agrado el contenido de este primer número. Por nuestra parte intentaremos mejorarlo mes a mes, con vuestra compañía seguro que lo conseguimos.

Hasta pronto.



Mundo MSX es una publicación del Grupo Editorial SYGRAN, S. A., Polig. Ind. Valdonaire, C/ Apolonia Hernández. HUMANES (Madrid). **Director:** Angel Herrero Fernández.
Director Adjunto: Andrés Franco. **Director Técnico:** Luis Sanguino. **Coordinador Software:** Angel García.
Secretaria Redacción: Mercedes Matons. **Diseño e Ilustraciones:** Javier Caballero. **Colaboradores:** Antonio García, Angeles Guindal, Carmen Torres.
Publicidad: GENESIS, Tomás López, 3, 6.°, 28009 Madrid, tfno: 401 77 54. Fotocomposición: Cuesta, S. A., Avda. Pedro Diez, 33 4.°.
Imprime: Graf. Osiris, S. A., Brañuelas, 29. Distribuye: G.M.E. Plaza Castilla, 3, Madrid. Depósito Legal: M - 31674 - 1987.

RECOMENDACIONES PARA TECLEAR UN PROGRAMA

Para aquellos que no estén muy acostumbrados a teclear programas daremos a continuación una serie de instrucciones o pasos a seguir para que les resulte más sencillo y las probabilidades de que se produzcan errores en la ejecución de los programas sean menores:

1. Como muchos ya sabrán las líneas que después de su número correspondiente tienen una instrucción REM o un apóstrofe ' no son consideradas por el programa sino que únicamente sirven como comentario aclaratorio para el programador y por lo tanto no es necesario que las copie completas cuando aparezcan en nuestros listados, en cambio sí es conveniente que ponga el número de línea y la instrucción REM o ' (ejemplo: 200 REM o 200 ') para evitar que puedan producirse errores al ejecutarse en el programa alguna instrucción de salto (GOSUB, GOTO, etc.).

2. Es fundamental que los números de línea los copie exactamente como figuran en el programa original. Esto puede realizarse fácilmente con la instrucción AUTO de la siguente forma: AUTO 10,10, donde el primer número 10 podrá ser sustituido por el número de línea en que reanudemos nuestra tarea, si no la terminamos de una sola vez.

3. Es conveniente, sobre todo en programas largos, salvar periódicamente a cinta o disco la parte que llevemos tecleada del listado antes de probar su funcionamiento para evitar la pérdida completa del programa en caso de que exista un error en el código máquina o bien se produzca un corte accidental de corriente, etc.

Dichas grabaciones periódicas es aconsejable no efectuarlas sobre la copia anterior o con el mismo nombre en caso de hacerlas en disco.



ESTE ES UN TRUCO CON EL QUE PODEIS CONTROLAR LA VELOCIDAD DE SONDEO DEL TECLADO

```
10 CLS:SCREEN 0:WIDTH 40:KEY OFF
20 LOCATE 2,2:PRINT"   TIENES UNOS SEGUNDOS PARA COMPROBAR LA VELOCIDAD NORMAL DE TU TECLADO, PARA ELLO MANTEN PULSADA UNA TECLA.
30 PRINT
40 X$=INKEY$
50 PRINT X$;
55 IF X$<>"" THEN I=I+1
60 IF I<100 THEN 40
70 LOCATE 2,10:PRINT"   COMPRUEBA AHORA COMO DICHA VELOCIDAD AUMENTA CONSIDERABLEMENTE VOLVIENDO A   MANTENER PULSADA UNA TECLA.
80 FOR J=1 TO 50:BEEP:NEXT
90 PRINT
100 POKE &HF3F6,1
110 POKE &HF3F7,1
120 X$=INKEY$
130 PRINT X$;
140 GOTO 100
```

COTILLEOS

MSX-III... SI, - MSX-III... NO



Han llegado a nuestra redacción rumores de que está siendo diseñado un nuevo compatible MSX con una capacidad gráfica muy superior a la de las versiones MSX-I y MSX-II.

El MSX-III cuya fecha de lanzamiento podría ser los primeros meses del año que viene, tendría como procesador principal un CHIP de MOTOROLA probablemente el mismo que incorpora el COMMODORE-AMIGA, por supuesto el microprocesador Z-80 también estaría presente en el nuevo MSX-III para asegurar su compatibilidad con los modelos anteriores de este mismo sistema.

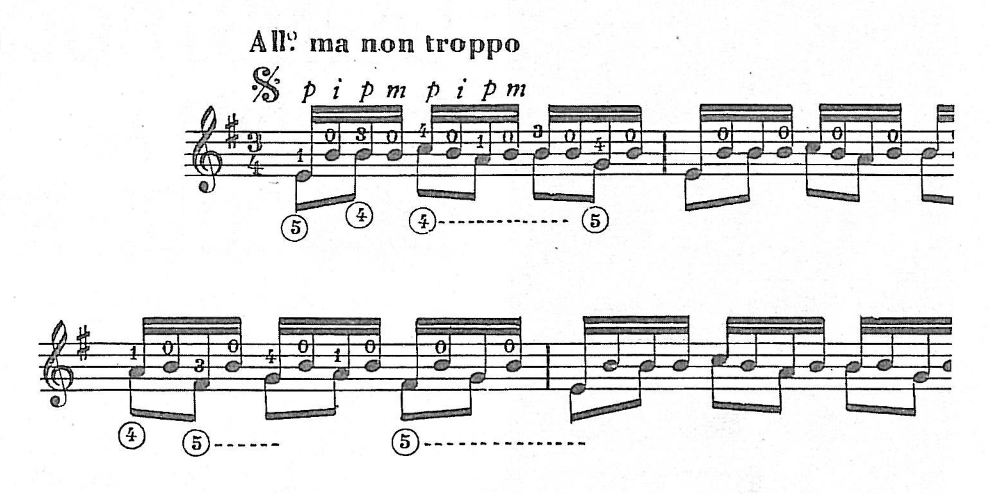
La redacción de Mundo-MSX se ha puesto en contacto con diferentes firmas fabricantes de ordenadores MSX para intentar que nos confirmaran este rumor y nos dieran una mayor información sobre las características que tendría el MSX-III. Ninguno de los fabricantes consultados ha admitido estar desarrollando este ordenador ni tener noticias de que nadie lo esté haciendo, sin embargo dado el gran secreto en que los fabricantes suelen envolver sus investigaciones hasta que no deciden comercializar un producto acabado, no sería de extrañar que lo que hoy aparece como simple rumor se convierta en breve en una noticia real y podamos disponer de un ordenador con capacidades gráficas casi impensables hasta ahora.

LA INSTRUCCION SOUND AL DESCUBIERTO



Todos los ordenadores suelen incorporar en sus circuitos, junto al procesador principal, que se encarga de realizar la mayor parte de las operaciones del sistema, uno o varios procesadores secundarios especializados en funciones muy concretas para las cuales están mejor dotados que el chip principal y además agiliza el trabajo de ésta, ya que le descarga de algunas tareas que, en caso de no realizarlas un procesador auxiliar, retardaría considerablemente sus acciones.

Así es frecuente encontrarnos en ordenadores de gestión que necesitan realizar numerosas operaciones matemáticas, coprocesadores aritméticos que realizan dichas operaciones mucho más rápidamente que el procesador central. Existen también coprocesadores de vídeo, sonido, etc.



En el caso de los ordenadores MSX, el procesador central o chip principal es como ya sabéis el Z80, que es ayudado en las funciones de vídeo y sonido por los procesadores auxiliares: el TMS 9192 A, que es el nombre técnico que recibe el procesador de imágenes de vídeo (VDP), y el AY-3-8910, más conocido como Generador Programable de Sonico (PSG).

Vamos a explicar de un modo sencillo y lo más ameno posible cómo funciona este Generador Programable de Sonido, en qué partes está dividido y cómo puede ser programado con la instrucción SOUND desde el MSX-BASIC.

El formato de la instrucción SOUND es el siguiente:

SOUND número de registro, dato a introducir.

Es decir, SOUND 8.10 cargará el registro número 8 del PSG con el valor 10.

Veamos ahora cómo funciona cada registro dentro del GENERADOR PROGRAMABLE DE SONIDO.

Existen en el PSG tres canales independientes y aunque esta independencia sea algo limitada como luego veremos, vamos a estudiarlos por separado para que os sea más fácil su comprensión. Empecemos, pues, por los registros pertenecientes al canal A o canal 1.

Sólo tres registros corresponden exclusivamente a cada canal, en nuestro caso los tres registros que el canal A tiene en exclusiva y que no comparte con los otros dos canales son el registro 0, y el registro 1 y el registro 8, cada uno de ellos cumple una función dentro del canal A que pasamos a explicar:

REGISTROS 0 y 1

Estos dos registros trabajan simultáneamente dentro del canal A y utilizan entre ambos 12 bits, los 8 de registro 0 y los 4 bits menos significativos del registro 1. Los 4 bits más altos de dicho registro no se usan (ver fig 1).

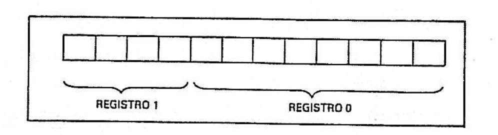


fig.1

Los doce bits de estos registros determinan el tipo de sonido que emitirá el canal A, como es lógico el contenido del registro 1, dado que ocupa los cuatro bits más significativos, dentro de los 12 utilizados por dicho canal A, tendrá una repercusión o influencia mayor en el tipo de nota a emitir que en el contenido del registro 0. De este modo, llamaremos al registro 1: **REGISTRO DE CONTROL DE GRAVES DEL TONO,** y al registro 0: **REGISTRO DE CONTROL DE AGUDOS DEL TONO.**

El valor más alto que puede contener el registro 0 es 11111111 binario, o lo que es lo mismo 255 decimal. Por su parte, el registro 1 sólo podrá almacenar un valor de 1111 binario, 15 en decimal.

Cuanto más alto sea el valor del conjunto de los 12 bits que forman estos dos registros, más bajo será el tono que generará el canal A.

Todavía no estamos en condiciones de emitir ningún sonido por nuestro canal, pues para ello necesitamos que los tonos que hemos programado en los registros 0 y 1 tengan una amplitud o volumen y esta es la función que realiza el registro 8.

REGISTRO 8

Es el registro de control de amplitud o volumen del canal A y utiliza para este cometid sus cuatro bits menos significativos; por tanto, el valor máximo que dicho registro deberá contener será 1111 binario (15 decimal). De este modo, 15 es el volumen o amplitud máxima que puede emitir el canal A siendo 0 el volumen mínimo.

Ya podemos ver algún ejemplo práctico de cómo funcionan los canales del PSG: Teclea SOUND 0,50:SOUND 8.15 y pulsa ENTER y oirás un pitido agudo constante y bastante alto, esto se debe a que hemos introducido un valor no demasiado alto en el registro 0, o registro de control de agudos de tono, y el volumen más alto que admite el registro 8, es decir, PI.

Podemos probar a ir modificando los valores de estos registros y veréis cómo cambia el tono y el volumen que el PSG emite. Si hacemos que el contenido de registro 0 sea igual a 0 haciendo SOUND 0,0 e introducimos lo siguiente: SOUND 1,10:SOUND 8.15 obtendremos un sonido mucho más grave, también podemos modificar los valores de ambos registros y comprobamos la amplia gama de tonos que un solo canal del PSG es capaz de emitir.

Hasta ahora sólo hemos hablado de un canal, el canal A, y ya habíamos dicho anteriormente que el generador de sonido de los ordenadores MSX posee tres canales idénticos pero independientes. Pues bien, la misma función que cumplen en el canal A los registros 0 y 1, lo llevan a cabo en el canal B los registros 2 y 3, es decir, el registro 2 sería el registro de control de agudos de tono del canal B, y el registro 3 sería el registro de control de graves de tono de dicho control.

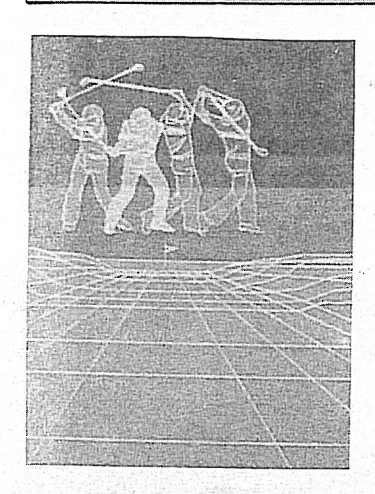


La amplitud o volumen del canal B será regulada por los valores que contengan el registro 9, que por supuesto funciona exactamente igual que el registro 8, para poder comprobar esto no tenéis más que introducir (una vez que hayáis silenciado, pulsando las telcas CTRL + STOP, cualquier sonido que estuviera emitiendo vuestro ordenador) a los registros del canal B los mismos valores que hemos dado antes a los registros pertenecientes al canal A. Por ejemplo, SOUND 2,50:SOUND 9.15 sonará exactamente igual que SOUND 0,50:SOUND 8.15.

Lo mismo sucede en el canal 3 o canal C, sólo que en este caso los registros que debemos utilizar serán 4 y 5 para control de tono y el 10 para control de volumen.

Las posibilidades del generador programable del sonido no se limitan sólo a la emisión de sonidos por los 3 canales, sino que también podemos programar otros de sus registros, de manera que el sólo se encargue de activar o desactivar uno u otro canal y también de elevar o disminuir automáticamente el volumen de cada uno de los 3 canales, siempre que el valor de los registros de control de amplitud de los canales, cuyo volumen queremos que sea controlado automáticamente por el ordenador, tengan valor 16, es decir, SOUND 8,16:SOUND 9.16 hará que el volumen de los canales A y B pueda ser controlado por el registro 13, por registro de control de envolventes del PSG, este registro así como el registro 7 que activa y desactiva los canales y el resto de registros que forman parte del generador programable de sonido, serán explicados con todo detalle en el próximo número de nuestra revista.

Animación por ordenador



Todos sabemos que los SPRITES de los ordenadores MSX no pueden tener más que un color y esto hace que los protagonistas de nuestros juegos resulten a menudo poco vistosos. Sin embargo la gran velocidad con que el BASIC-MSX "mueve" estas figuras por la pantalla, nos permite combinar varias de ellas de manera que entre todas formen un único personaje animado y de varios colores.

Nuestro primer protagonista va a ser un atleta que correrá ágilmente según vayamos pulsando la barra espaciadora de nuestro ordenador.

Para que la pantalla de fondo resulte más vistosa hemos dibujado unas gradas y una pista simulando un estadio mediante la redefinición de una serie de caracteres. Como podréis apreciar observando el listado, las rutinas que se encargan de redefinir estos caracteres y colocarlos en pantalla se encuentran entre la línea 50 y la 480, y los datos que utilizan están entre las líneas 1340 y 1470. Estas rutinas y sus datos únicamente sirven como decorado y por lo tanto no afectan para nada al programa de animación que nos ocupa, por lo tanto aquellos que tengáis prisa por ver a nuestro protagonista en acción podéis no teclear las líneas anteriormente mencionadas respetando, eso sí, los números de línea que ocupan, es decir, si decidís no copiar las líneas comprendidas entre la 50 y la 480 tendréis obligatoriamente que suprimir también sus datos (líneas 1340-1470). Pero la rutina de formación de sprites deberá seguir comenzando en la línea 510. No sucede los mismo con los datos que pueden encontrarse a partir de cualquier número de línea con la única condición de que se respete rigurosamente su orden.

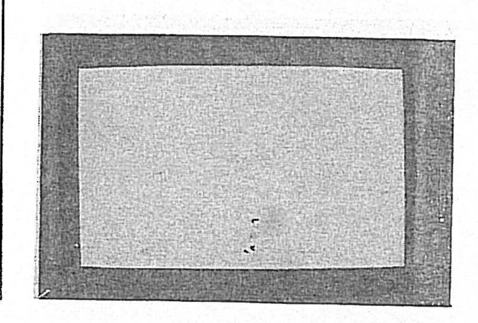
```
10 RE=99
20 COLOR 1,5,1:RESTORE
30 SCREEN 1,2,0:KEY OFF:WIDTH 32
40 '
50 'OU REDEFINICION DE CARACTERES OO
60 '
70 '
80 '           GRADA SUPERIOR
90 '
100 FOR N=0 TO 31:VPOKE 6272+N,220:NEXT
110 'VPOKE 9992,25
120 '
130 '               PUBLICO
140 '
150 FOR I=1792 TO I+15:READ A$
160 VPOKE I,VAL("&H"+A$):NEXT
170 FOR N=32 TO 159 STEP 2:VPOKE 6272+N,
224:VPOKE 6273+N,225:NEXT
180 VPOKE8220,118
190 '
200 '           GRADA INFERIOR
210 '
220 FOR N=160 TO 191:VPOKE 6272+N,219:NE
XT
230 '
240 '             LADRILLOS
250 '
260 FOR I=1856 TO I+15:READ A$
270 VPOKE I,VAL("&H"+A$):NEXT
280 FOR N=192 TO 318 STEP 2:VPOKE 6272+N
,232:VPOKE 6273+N,233:NEXT
290 FOR I=352 TO I+7:READ A$
300 VPOKE I,VAL("&H"+A$):NEXT
310 VPOKE 8192+29,25
320 LOCATE 12,11:PRINT"BEBA,ROCA-MOLA"
330 '
340 '               PISTA
350 '
360 FOR I=1496 TO I+15:READ A$
370 VPOKE I,VAL("&H"+A$):NEXT
```

```
380 FOR N=320 TO 351 STEP 2:VPOKE 6272+N
,188:VPOKE 6273+N,187:NEXT
390 VPOKE 8192+23,252
400 FOR I=1984 TO I+15:READ A$
410 VPOKE I,VAL("&H"+A$):NEXT
420 VPOKE 8192+31,159
430 FOR N=352 TO 447 STEP 2:VPOKE 6272+N
,248:VPOKE 6273+N,249:NEXT
440 FOR N=448 TO 479 STEP 2:VPOKE 6272+N
,187:VPOKE 6273+N,188:NEXT
450 FOR N=480 TO 575 STEP 2:VPOKE 6272+N
,248:VPOKE 6273+N,249:NEXT
460 FOR N=576 TO 639 STEP 2:VPOKE 6272+N
,187:VPOKE 6273+N,188:NEXT
470 FOR I=1 TO 10:BEEP:NEXT
480 LOCATE 12,3:PRINT"LISTO!!!"
490 '
500 '
510 'OOOO FORMACION DE SPRITES OOOO
520 '
530 FOR K=1 TO 11:B$="":FORI=1TO 32 :REA
D A$:B$=B$+CHR$(VAL("&H"+A$)):NEXT:SPRIT
E$ (K)=B$:NEXT
540 FOR K=12 TO 15:B$="":FORI=1TO 8 :REA
D A$:B$=B$+CHR$(VAL("&H"+A$)):NEXT:SPRIT
E$ (K)=B$:NEXT
550 '
560 '
570 '          DISPARO DE SALIDA
580 '
590 SOUND 6,15:SOUND 7,7:SOUND 8,16:SOUN
D 9,16:SOUND 10,16:SOUND 11,0 :SOUND 12,
16:SOUND 13,0
600 '
610 TIME=0:A=5:B1=4
620 '
630 LOCATE 12,3:PRINT"        "
640 GOSUB 1100
650 '
660 ' SONDEO DE LA BARRA ESPACIADORA
670 '
680 ON STRIG GOSUB 770:STRIG(0) ON
690 '
700 '  OOOO BUCLE PRINCIPAL OOOO
710 '
720 '
730 IF A>240 THEN GOTO 840
740 TI=TIME/50
750 GOTO 720
760 '
770 A=A+1.2:B1=B1+1.38:VC=(VC+1)MOD(3)+1
780 SWAP A,B1
790 ON VC GOSUB  1100,1180,1260
```

Como podéis apreciar nuestro programa está realizado en el modo o SCREEN 1 únicamente para poder introducir los caracteres que forman el decorado y los textos auxiliares, de otro modo podríamos haber utilizado el modo gráfico de alta resolución (SCREEN 2).

El sistema que hemos utilizado para dar animación a nuestro atleta es similar a como se hace un dibujo animado en una película. Es decir, nuestro personaje consta de tres dibujos o grupos de SPRITES, cada uno de los cuales corresponde a una postura diferente, que aparecen sucesivamente en la pantalla sustituyendo al anterior cada vez que se pulsa la barra espaciadora. Por lo tanto el bucle principal de nuestro programa únicamente tendrá que determinar qué figura debe aparecer en cada momento e ir incrementando la coordenada horizontal de la misma a cada pulsación de la barra espaciadora.

Para que la velocidad de movimiento de nuestro protagonista dependa únicamente del número de pulsaciones



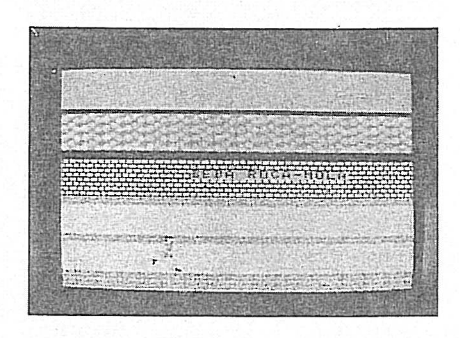
sobre el espaciador y no se vea afectada o entorpecida por la ejecución de otras rutinas del programa en la línea 680, hemos activado un sondeo por interrupciones de dicho espaciador, el cual hace que el ordenador acuda varias veces por segundo a comprobar si esta tecla ha sido pulsada y vuelta a soltar.

EXPLICACION DE LAS LINEAS DEL LISTADO

— De la línea 10 a la 30 se inicializa la variable RE (récord de tiempo) y se escoge el color y modo de pantalla, así como se reactivan las líneas de datos (RESTORE) para que puedan volver a ser leídos por el programa en caso de jugarse varias partidas.

— Como ya hemos dicho entre las líneas 50 y 510 se confecciona la pantalla de fondo.

— Entre las líneas 510 a la 540 se forman todos los sprites que configuran a nuestro atleta a partir de los datos que comienzan en la línea 1500.



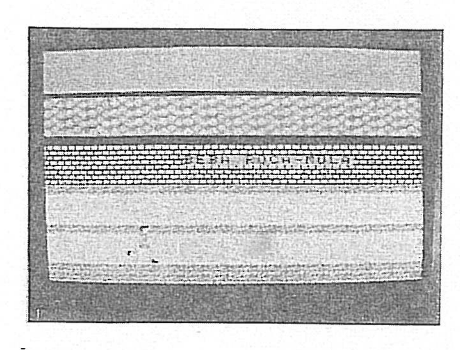
```
800 RETURN
810 '
820 '  RUTINA DE LLEGADA A LA META
830 '
840 STRIG(0) OFF
850 SOUND 6,40:SOUND 7,55:SOUND 8,7:SOUN
D12,45:W=9:TIME=0
860 FOR I=99 TO 231
870 VPOKE 8220,I
880 FOR J=1 TO 20:NEXT
890 NEXT
900 VPOKE8220,118
910 SOUND 8,0
920 FOR I=1 TO 500:NEXT
930 '
940 IF TI<RE THEN RE=TI
950 LOCATE 2,1:PRINT"TIEMPO INVERTIDO:";
TI;"SEG."
960 LOCATE 2,2:PRINT"MEJOR TIEMPO....:";
RE;"SEG."
970 LOCATE 2,3:PRINT"OTRA PARTIDA SI O N
O [S/N]"
980 X$=INKEY$
990 IF X$="S" OR X$="s" THEN GOTO 20
1000 IF X$="n" OR X$="N" THEN SCREEN 0:E
ND
1010 GOTO 980
1090 '
1100 PUTSPRITE 1 ,(A,144), 1, 1
1110 PUTSPRITE 2,(A,144),10, 3
1120 PUTSPRITE 3 ,(A-1,158),4,6
1130 PUTSPRITE 4 ,(A,144),6, 2
1140 PUTSPRITE 5,(A-1,157),10,9
1150 PUTSPRITE 6,(A-9,22+142),1,12
1160 PUTSPRITE 7,(A+10,29+142),1,13
1170 RETURN
1180 PUTSPRITE 1 ,(A,144), 1, 1
1190 PUTSPRITE 2,(A,144),10, 4
1200 PUTSPRITE 3 ,(A-1,158),4,7
1210 PUTSPRITE 4 ,(A,144),6, 2
1220 PUTSPRITE 5,(A-1,157),10,10
1230 PUTSPRITE 6,(A-5,30+142),1,14
1240 PUTSPRITE 7,(A-1,27+142),1,15
1250 RETURN
1260 PUTSPRITE 1 ,(A,144), 1, 1
1270 PUTSPRITE 2,(A,144),10, 5
1280 PUTSPRITE 3 ,(A-1,158),4,8
1290 PUTSPRITE 4 ,(A,144),6, 2
1300 PUTSPRITE 5,(A-1,157),10,11
1310 PUTSPRITE 6,(A-4,23+142),1,12
1320 PUTSPRITE 7,(A+1,31+142),1,13
1330 RETURN
1340 '          DATOS PUBLICO
```

```
1350 '
1360 DATA 1,2,33,79,B4,FC,48,30
1370 DATA C8,A0,E0,48,9C,2A,3E,14
1380 '
1390 '        DATOS LADRILLOS
1400 '
1410 DATA42,42,42,FF,10,10,10,FF,8,8,8,F
F,42,42,42,FF
1420 DATA 0,0,0,0,0,0,0,0
1430 '
1440 '          DATOS PISTA
1450 '
1460 DATA24,40,1,24,92,0,8A,11,89,0,81,8
,42,0,14,49
1470 DATA 21,10,1,24,40,0,82,1,0,4a,21,0
,42,0,14,49
1480 '
1490 '
1500 'OOOO DATOS DE LOS SPRITES OOOO
1510 '
1520 '
1530 '       DATOS CABEZA (S-1)
1540 '
1550 DATA 3,7,6,6,2,0,0,0,0,0,0,0,0,0,
0,C0,E0,80,0,0,0,0,0,0,0,0,0,0,0,0
1560 '
1570 '      DATOS CAMISETA (S-2)
1580 '
1590 DATA 0,0,0,0,0,0,3,7,7,7,7,7,7,7,7,
3,0,0,0,0,0,0,80,C0,C0,C0,C0,C0,C0,C0,80
,0
1600 '
1610 ' DATOS CARA Y BRAZO-1 (S-3)
1620 '
1630 DATA 0,0,1,1,1,1,1,0,1,3,6,C,6,3,1,
0,0,0,40,E0,C0,80,0,0,80,80,0,0,0,C0,C0,
C0
1640 '
1650 ' DATOS CARA Y BRAZO-2 (S-4)
1660 '
1670 DATA 0,0,1,1,1,1,1,0,1,1,1,1,0,0,0,
0,0,0,40,e0,c0,80,0,0,80,80,Fc,Fc,6,0,0,
0
1680 '
1690 ' DATOS CARA Y BRAZO-3 (S-5)
1700 '
1710 DATA 0,0,1,1,1,1,1,0,1,1,1,1,0,0,0,
0,0,0,40,E0,C0,80,0,C,9C,B0,E0,80,0,0,0,
0
1720 '
1730 ' DATOS  PANTALON-1 (S-6)
1740 '
1750 DATA 1,3,7,7,3,0,0,0,0,0,0,0,0,0,0,
```

— A partir de la línea 570 y hasta la 640 se produce un sonido que emula un disparo de salida y se pone el cronómetro a cero, inicializándose también las variables A y B1 que determinarán la coordenada horizontal de nuestro protagonista. Posteriormente se borra la palabra "LISTO!!!" y se hace aparecer por primera vez a nuestra figura en la pantalla con su primera postura (GOSUB 1100).

— Como ya hemos dicho la línea 680 sondea la barra espaciadora y a partir de la línea 700 comienza el pequeño bucle principal de nuestro programa que únicamente se encarga de comprobar si hemos llegado a la meta (línea 730) y el tiempo que invertimos (línea 740).

— En la línea 770 y hasta la 800 hemos incluido una rutina que no pertenece a lo que hemos llamado bucle principal, pero que en realidad es la encargada de gestionar todos los movimientos de nuestro atleta. A esta rutina acude el programa cada vez que es pulsada la barra espaciadora y en ella se incrementan las variables A y B1 y Vc o Vc determina cuál



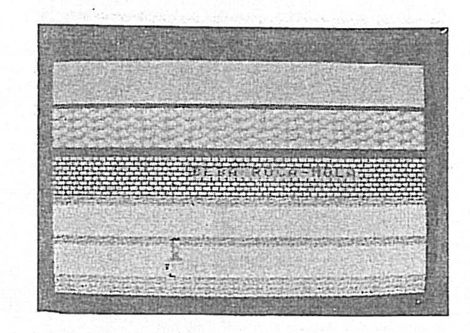
de las tres figuras que constituyen nuestro sprite debe aparecer en pantalla, por eso lleva un operador (MOD (3)) que evita que VC tome un valor mayor que 3.

— El incremento de las variables A y B1 es como podemos apreciar muy parecido, pero ligeramente diferente; esto lo hemos hecho así para que la línea 780 que sustituye una por otra obligue a nuestro personaje a realizar una pequeña vibración, la cual da mayor realismo a su movimiento. Por último la línea 790 desvía el programa hacia la rutina que representa la postura correcta del corredor.

— Entre las líneas 820 hasta la 1010 hemos introducido una serie de comentarios que aparecen cuando se llega a la meta.

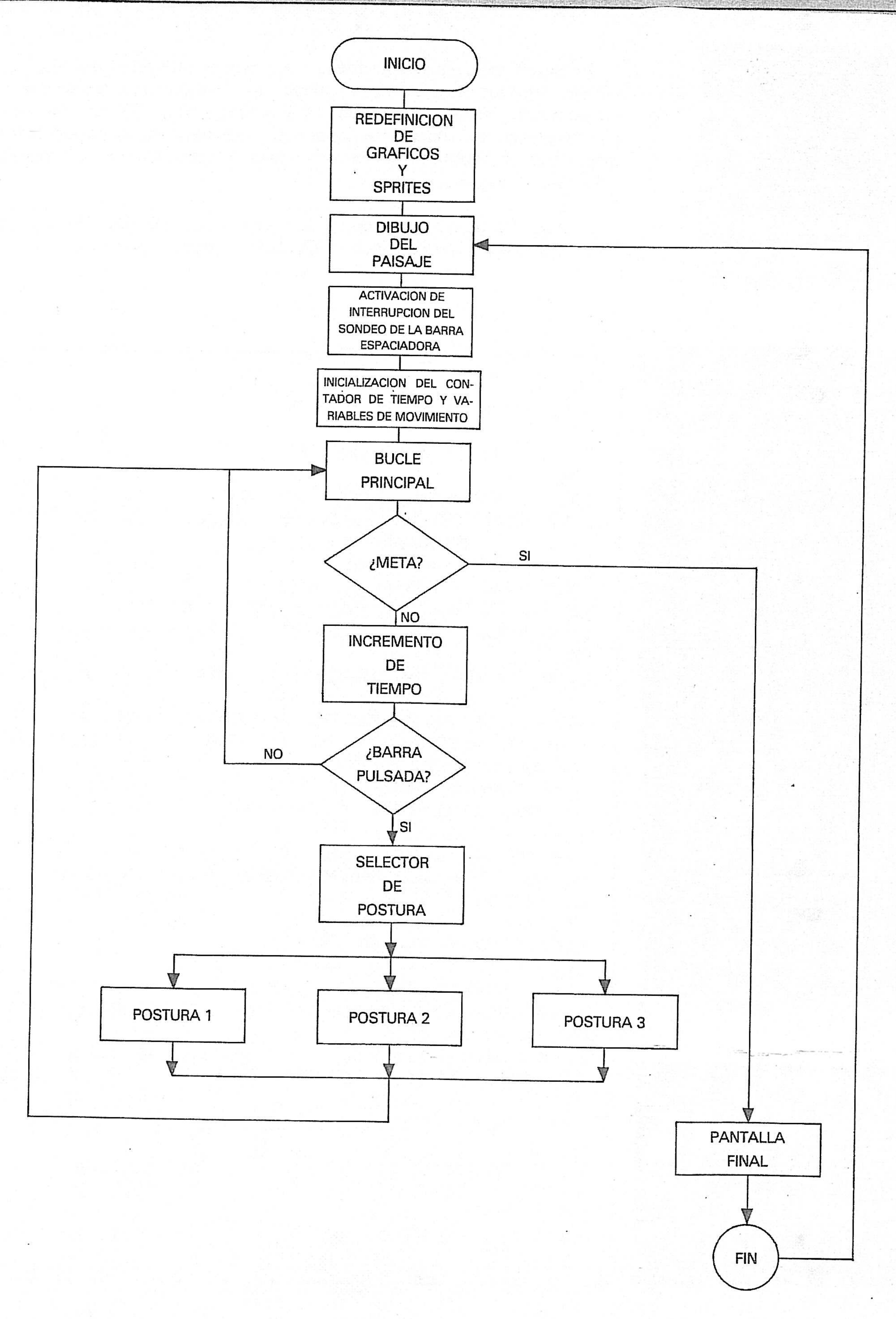
— En la línea 1100 comienzan los tres grupos de instrucciones que hacen aparecer en pantalla al protagonista de nuestro juego en tres posturas diferentes; el primer grupo o primera postura va de la línea 1100 a la 1170, el segundo de la 1180 a la 1250 y el tercero de la 1260 a la 1330.

```
0,C0,E0,F0,70,60,0,0,0,0,0,0,0,0,0,0
1760 '
1770 ' DATOS PANTALON-2 (S-7)
1780 '
1790 DATA 3,7,F,F,F,8,0,0,0,0,0,0,0,0,0,
0,C0,C0,E0,F0,C0,0,0,0,0,0,0,0,0,0,0
1800 '
1810 ' DATOS PANTALON-3 (S-8)
1820 '
1830 DATA 1,3,7,7,3,0,0,0,0,0,0,0,0,0,0,
0,E0,F0,F8,F8,70,0,0,0,0,0,0,0,0,0,0
1840 '
1850 ' DATOS PIERNAS-1 (S-9)
1860 '
1870 DATA 0,0,0,0,C,E,E,FC,F8,0,0,0,0,0,
0,0,0,0,0,0,0,78,3c,1e,C,C,C,C,C,C,0,0
1880 '
1890 ' DATOS PIERNAS-2 (S-10)
1900 '
1910 DATA 0,0,0,0,0,7,7,7,7,7,E,1C,18,30
,60,40,0,0,0,0,30,78,3C,1C,38,70,60,C0,0
,0,0,0
1920 '
1930 ' DATOS PIERNAS-3 (S-11)
1940 '
1950 DATA 0,0,1,1,1,1,1,1,7,F,9,1,3,6,6,
C,0,0,F8,FC,FE,DF,CF,FE,FC,C0,80,80,0,0,
0,0
1960 '
1970 ' DATOS ZAPATOS-1 (S-12 Y 13)
1980 '
1990 DATA 7,7,6,6,4,0,0,0,30,3C,3E,0,0,0
,0,0
2000 '
2010 ' DATOS ZAPATOS-2 (S-14 Y 15)
2020 '
2030 DATA 4,C,E,7,0,0,0,0,1,7,6,3,3,0,0,
0
2040 '
```

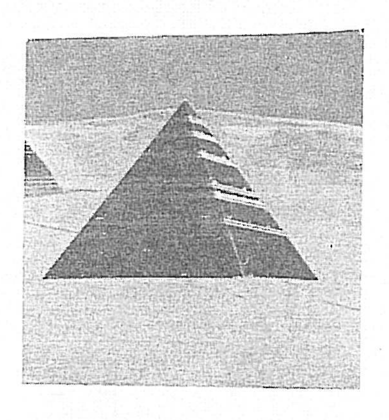


SUGERENCIAS

Es posible modificar la velocidad y la vibración del corredor modificando los incrementos de las variables A y B1 en la línea 770. También podéis cambiar los colores del atuendo de nuestro atleta cambiando los atributos de color de las líneas 1100 a la 1320. Y como reto a vuestros conocimientos de programación podéis también hacer que nuestro protagonista se enfrente contra otro corredor que maneje el ordenador, para ello podéis usar los mismos números de SPRITES y colocarlos en la otra pista de nuestro decorado formando un contrincante.


INICIO
REDEFINICION DE GRAFICOS Y SPRITES
DIBUJO DEL PAISAJE
ACTIVACION DE INTERRUPCION DEL SONDEO DE LA BARRA ESPACIADORA
INICIALIZACION DEL CONTADOR DE TIEMPO Y VARIABLES DE MOVIMIENTO
BUCLE PRINCIPAL
¿META?
SI
NO
INCREMENTO DE TIEMPO
¿BARRA PULSADA?
NO
SI
SELECTOR DE POSTURA
POSTURA 1
POSTURA 2
POSTURA 3
PANTALLA FINAL
FIN

KEOPS (Construye la pirámide de Keops):



Tu misión en este juego consiste en construir en un tiempo limitado, y con los ladrillos dispersos por la zona una pirámide, pero previamente tendrás que atravesar los túneles malditos esquivando las flechas que surgen de los agujeros de las paredes y procurando no tocar tampoco al guardián del túnel, ya que sus púas están cargadas de un fuerte veneno que sin duda te producirá la muerte. Dispones de 5 vidas para conseguirlo.

El programa ha sido grabado en tres bloques: un cargador, los datos de los sprites (generados con el programa MGEM), y el programa principal.

Pasos a seguir para grabar el programa en cinta:

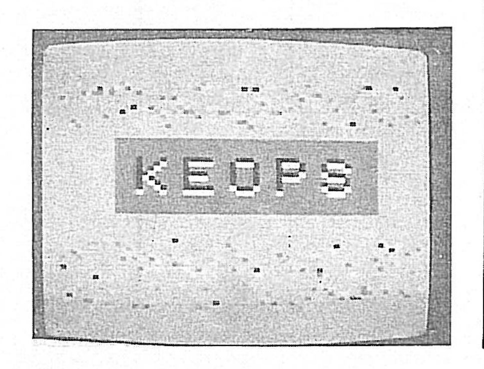
1. Teclear el programa cargador y grabar en cinta con: SAVE "CAS: KEOPS".

2. Teclear el programa principal. Hacer RUN 60000.

3. Subir el cursor hasta la palabra BSAVE y pulsar RETURN. Este programa se grabará a continuación del cargador.

El programa principal adopta la siguiente estructura:

1000-1270: Dibujo de la pantalla de presentación y elección de la forma de control.




```
1000 '
1010 ' ** KEOPS * A.G.D. *
1020 '
1030 ' **  PRESENTACION **
1040 '
1050 CLEAR1000,57999!:KEYOFF
1060 BEEP:DEFINT A-Z:SCREEN3,2,0:COLOR1,
7,7:POKE&HFCAB,1:CLS
1070 OPEN"GRP:"AS#1
1080 LINE(48,58)-(228,118),3,BF
1090 LINE(40,50)-(220,110),12,BF
1100 COLOR 15:PSET(56,70),7:PRINT#1,"KEO
PS"
1110 COLOR1:PSET(56,66),7:PRINT#1,"KEOPS
"
1120 PLAY"O4T200CDEFEDEFAFEDEFA.CDCDCDEF
EDEFADEFAFEDCDCCC","O6T200CDEFEDEFAFEDEF
A.CDCDCDEFEDEFADEFAFEDCDCCC"
1130 FORA=0TO255STEP2:PSET(A,195-RND(1)*
60),RND(1)*15:PSET(255-A,RND(1)*40),RND(
1)*15:NEXTA
1140 FORA=1TO1000:NEXTA:FORA=0TO260STEP4
:LINE(A,0)-(A,196),8:LINE(A-4,0)-(A-4,19
6),7:NEXTA
1150 '
1160 ' ** FORMA DE CONTROL **
1170 '
1180 SCREEN0:COLOR1,7,7:CLS
1190 LOCATE5,4:PRINT"*** LA PIRAMIDE DE
KEOPS ***"
1200 PRINT:PRINT"          1986 GENESIS SO
FTWARE":PRINT:PRINT"                A.G.
D."
1210 LOCATE 7,13:PRINT"ELIGE FORMA DE CO
NTROL:"
1220 PRINT:PRINT"              1. TECLADO"
1230 PRINT:PRINT"              2. JOYSTICK
 A"
1240 PRINT:PRINT"              3. JOYSTICK
 B"
```

```
1250 A$=INKEY$
1260 IFA$>"0"ANDA$<"4"THENJ=VAL(A$)-1 EL
SE 1250
1270 '
1280 ' ** INICIO PROGRAMA **
1290 '
1300 BEEP:SCREEN1,2,0
1310 DEFUSR0=58000!:COLOR1,2,2:CLS
1320 P=0:V=5:A=RND(-TIME)
1330 '
1340 ' ** FASE 1 **
1350 '
1360 LOCATE12,7:PRINT"FASE 1:":PRINT:PRI
NT:PRINT:PRINT"      LOS TUNELES MALDITOS
":PRINT:PRINT:PRINT:PRINT"        250 PUNT
OS x TRAMO":GOSUB2720
1370 GOSUB2660
1380 C=0
1390 SCREEN,2:A=USR0(0)
1400 C=C+1:IFC>6THEN1920
1410 BEEP:SPRITEOFF:FORA=0TO22:LOCATE0,A
:PRINT"çççççççêè              èêçççççç";:N
EXTA
1420 X=122:Y=2:M=1
1430 PUTSPRITE0,(X,Y),1,0
1440 F=1+INT(RND(1)*2):ONFGOSUB1640,1770
1450 SPRITEOFF:ONSPRITEGOSUB1830
1460 '
1470 ' ** BUCLE JUEGO **
1480 '
1490 D=STICK(J)
1500 IFD=8ORD=1ORD=2THENY=Y-4:IFY<2THENY
=2
1510 IFD=6ORD=5ORD=4THENY=Y+4:IFY>170THE
NP=P+250:CLS:GOTO1390
1520 IFD=2ORD=3ORD=4THENX=X+4:IFX>160THE
NX=160
1530 IFD=8ORD=7ORD=6THENX=X-4:IFX<90THEN
X=90
1540 SPRITEOFF
1550 PUTSPRITE0,(X,Y),1,M-1
1560 ONFGOSUB1650,1780
1570 M=3-M
1580 SPRITEON
1590 GOTO1490
1600 '
1610 ' ** SUBRUTINAS **
1620 ' ** MOVIMIENTO **
1630 '
1640 F1=0:F2=0:RETURN
1650 IFF1=0ANDRND(1)>.7THENF1=4:X1=80:Y1
=20+INT(145*RND(1)):GOSUB2760
1660 IFF1=0THEN1690
```

1280-1330: Inicialización de variables.

1340-1960: Fase 1 del juego, que consta de las siguientes partes:

— Inicialización 1360-1450.

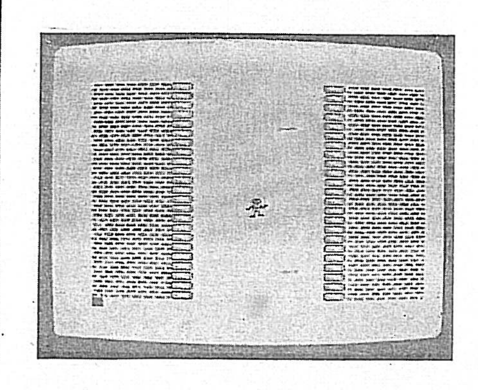
— Bucle del juego 1470-1590.

— Subrutinas de movimiento 1600-1820.

— Destruido 1840-1910.

1970-2650: Fase 2 del juego, con las partes:

— Dibujo escenario 2000-2120.



— Bucle del juego 2140-2340.

— Movimiento 2360-2450.

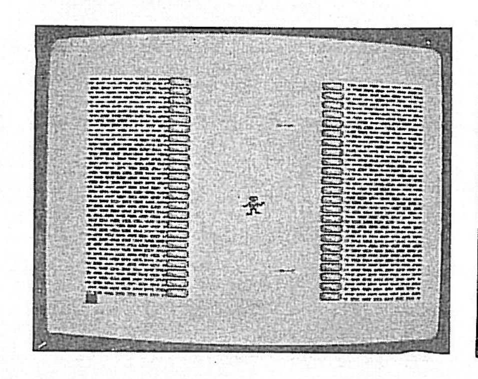
— Acción de coger o soltar un ladrillo 2460-2540.

— Comprobación de colisiones 2560-2590.

— Muerte 2600-2650

2670-2710: Subrutina que redefine los caracteres y les da color.

2720-2790: Varias subrutinas:



```
1670 X1=X1+8:IFX1>168THENF1=0
1680 PUTSPRITE1,(X1,Y1),F1,2
1690 IFF2=0ANDRND(1)>.7THENF2=4:X2=168:Y
2=20+INT(145*RND(1)):GOSUB2760
1700 IFF2=0THEN1730
1710 X2=X2-8:IFX2<82THENF2=0
1720 PUTSPRITE2,(X2,Y2),F2,3
1730 IFY1>Y2-8ANDY1<Y2+8THENY2=Y2-4:GOTO
1730
1740 RETURN
1750 '
1760 '
1770 SOUND8,10:SOUND7,56:X1=120:Y1=160:F
1=4:RETURN
1780 Y1=Y1-10+ABS(F1):X1=X1+F1:IFX1>1500
RX1<90THENF1=-F1
1790 IFY1<0THENY1=160:F1=2+INT(8*RND(1))
1800 PUTSPRITE1,(X1,Y1),2,M+3
1810 SOUND0,X1
1820 RETURN
1830 '
1840 ' ** MUERTO 1 **
1850 '
1860 SPRITEOFF:GOSUB2770:FORA=1TO12:PUT
SPRITE0,(X,Y),1,A:FORB=1TO250:NEXTB:NEXT
A:FORB=1TO2000:NEXTB
1870 V=V-1
1880 COLOR 15,13,13:FORA=0TO3:PUTSPRITEA
,(16*A,16*A),0:NEXTA:CLS
1890 LOCATE8,8:PRINT"PUNTOS...:";P:LOCAT
E8,12:PRINT"VIDAS....:";V
1900 IFV<1THENRETURN2790
1910 FORA=1TO2000:NEXTA:COLOR1,10,10:CLS
:GOSUB2710:RETURN 1410
1920 SCREEN,2:CLS:COLOR1,9,9
1930 LOCATE5,10:PRINT"BONO DE 1000 PUNTO
S"
1940 PLAY "O4CDEFDEC","O6CDEFDEC","O2CDE
FDEC"
1950 IFPLAY(0)=-1THEN1950
1960 CLS
1970 '
1980 ' ** FASE 2 **
1990 '
2000 COLOR1,2,2:CLS
2010 LOCATE12,7:PRINT"FASE 2:":PRINT:PRI
NT:PRINT:PRINT"      CONSTRUYE LA PIRAMIDE
":PRINT:PRINT:PRINT:PRINT"       50 PUNTOS
 x LADRILLO":GOSUB2720
2020 A=USR0(0):GOSUB2660
2030 CLS:PRINT"ççççççççççççççççççç
çççççç PT:        ç     ç TI:      çççççççç
ççççççççççççççççç";
```


(Continua en pág. 21)

LO ULTIMO DE KONAMI

QBERT

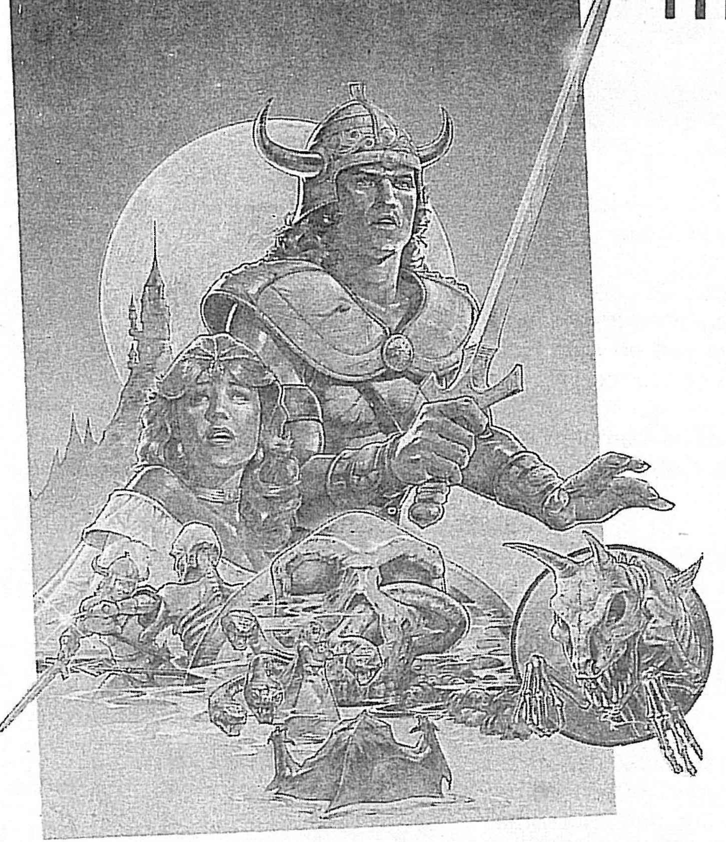


Salta sobre los cubos de colores y hazlos girar. Alinea los cinco cubos superiores en una fila para obtener una gran puntuación. QBERT: no te confíes..., parece fácil, pero es complicado. Es complicado pero muy divertido esta es la primera aparición de la mascota de KONAMI, figura favorita WRAPPY. WRAPPY realiza los puzzles de acción más costosos del mundo.

FORMATO: CARTUCHO ROM

BUSCALO... PROXIMAMENTE...

THE MAZE OF GALIOUS

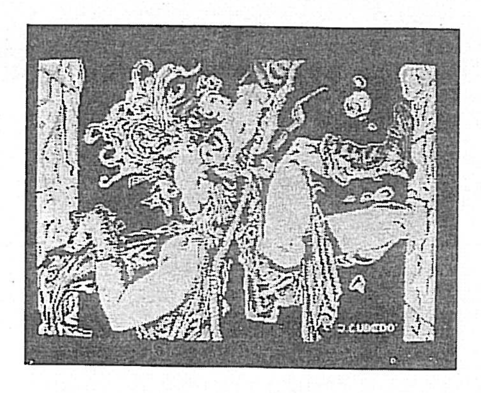


El castillo mágico era en realidad una trampa creada por el perverso Sumo Sacerdote GALIOUS. La bella AFRODITA fue capturada por el oscuro caballero HUDNOS. POPOLON parte hacia el monte ATLAS, donde ella estaba presa, no sabiendo que iba a caer en una trampa. GALIOUS, aprovechando que POPOLON estaba ausente, toma CASTLE GREEK y luego acude a los CIELOS rapta al todavía no nacido PAMPAS, quien nacería de POPOLON y AFRODITA. La pareja, a su regreso del monte ATLAS, descubre lo sucedido y con determinación atraviesa las puertas del castillo...

FORMATO: CARTUCHO ROM



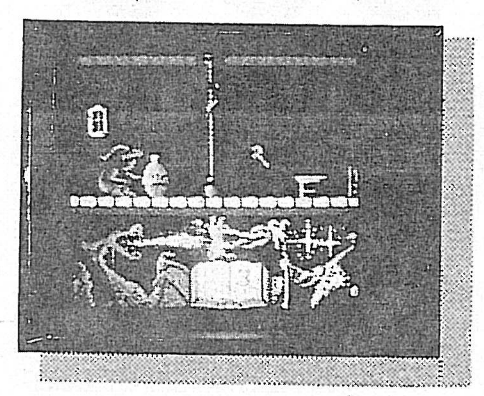
NONAMED



VIDEO-AVENTURA
DINAMIC

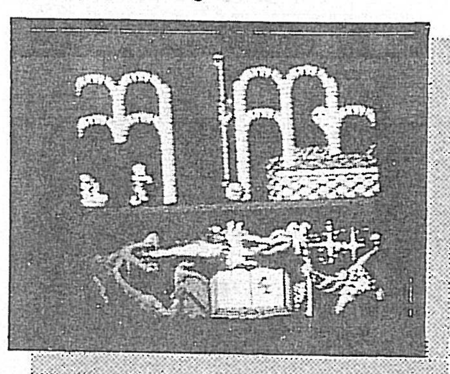
Erase que se era una época pasada donde los hombres para demostrar su valor y conseguir ser nombrados Caballeros del Rey, debían someterse a las más duras de las pruebas.

En un valle olvidado de la Baja Sajonia existía un pueblo de guerreros y campesinos que rendía pleitesía a su anciano y venerable monarca, el rey ABDUL HONEICKAM GARGOY.



Los jóvenes aspirantes a Caballeros de su Majestad eran seleccionados en todas las comarcas y villas de entre aquéllos que destacaban en el manejo del arco, la lucha a cuerpo o la espada.

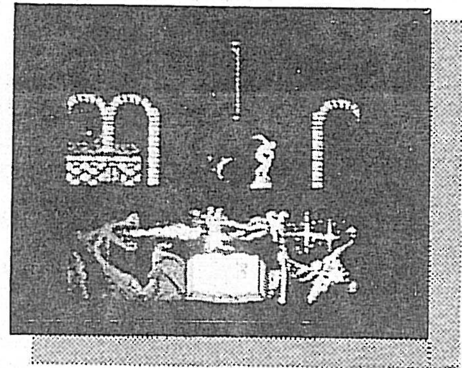
Sin embargo, el más duro de los sacrificios quedaba reservado sólo a unos pocos elegidos.



Uno por uno eran sometidos al rito final, el rito del Castillo sin nombre.

Consistía la prueba en ser encerrado en el Castillo sin nombre, intentar sobrevivir al mundo de pesadilla que existía dentro de sus muros y escapar con vida. Para ello la cuestión fundamental era encontrar la puerta.

Muchos jóvenes aspirantes a caballeros intentaron el reto, pero la gran mayoría nunca consiguieron sobrevivir y fueron convertidos en ogros bajo los poderosos hechizos



que el malvado mago NILREM reservaba a los perdedores.

Para escapar con vida del Castillo sin nombre, deberás entregar algo al mago NILREM que en agradecimiento te pagará con un conjuro.

Deberás impedir el contacto con los ogros, en caso contrario tu suerte será fatal.

Otra de las claves del éxito tiene relación con los jardines interiores del castillo en donde se encuentra la Fuente Sagrada, su agua te dará mucha fuerza y hará más vigorosos tus saltos.

Las capacidades físicas del protagonista son variadas, puede correr para escapar de sus perseguidores, golpearles a base de patadas, trepar por cuerdas y agacharse para no ser herido por los enemigos aéreos.

Los movimientos por lo general son rápidos, aunque el scroll lateral con el que se desplaza el paisaje no sea muy suave.

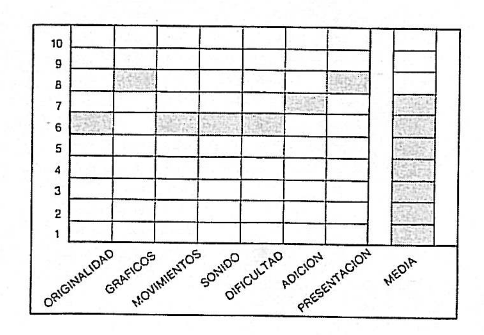
En lo referente al sonido, una melodía acorde con el tema, acompaña el menú de opciones. En cambio, durante el juego el sonido es casi nulo, sólo se escuchan las pisadas de nuestro héroe.

La realización de los gráficos está muy trabajada, de hecho el marcadorr de las vidas ocupa «un tercio de la pantalla». Sin embargo no aparece una tabla de puntuación que anime al jugador a superarse.

Hay poca diversidad en los tipos de enemigos, que son los mismos durante todo el juego.

La dificultad es elevada, pero no en exceso. Esto unido al deseo del jugador por conocer nuevas pantallas, hacen de NONAMED un juego adictivo.

La idea de la aventura no es muy original, pero la presentación ha sido cuidada en todos sus detalles.



SURVIVOR

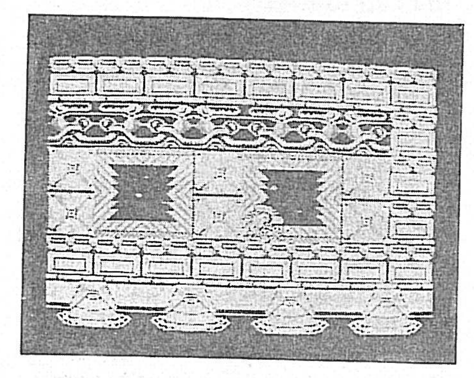
VIDEO-AVENTURA TOPO SOFT

En algún lugar del espacio, girando en torno a un viejo planeta ya casi muerto, flota una inmensa astronave cargada con todas las formas de vida de un millar de mundos. La nave es oscura, extraña y poderosa, y dispone de energías capaces de desgarrar la propia textura del espacio. Sus amos son dueños de secretos tecnológicos más allá de toda imaginación, y su palabra es ley para miles de razas en toda la galaxia. Y sin embargo, estos seres tan poderosos tienen un terrible problema...!

En algún lugar de la enorme estructura, arrastrándose sinuosamente por algún conducto de ventilación, recorriendo los invernaderos y salas de control, o deslizándose por los transportadores, una solitaria criatura lucha por su supervivencia y por la de su raza. La criatura es el producto final de un millón de años de evolución sobre un mundo agonizante. Astuta y llena de instintos salvajes, con músculos resistentes como el metal y ojos duros como el diamante, se resiste a los altivos dominadores.

El objetivo es perpetuar la raza introduciendo diez vainas en las incubadoras repartidas por toda la nave. El alienígena dispone de una energía que se irá agotando poco a poco. La única forma de reponer energía es soltando las vainas en las incubadoras, o bien devorando a los nanoingenieros que hay por la nave. La criatura puede defenderse escupiendo ácido paralizante o escondiéndose en conductos de ventilación, etc.

La nave consta de 142 habitáculos repartidos en cuatro planos que se unen a través de las puertas o de las rejillas de ventilación. Cada plano tiene varios niveles a los que se accede por medio de ascensores. Para

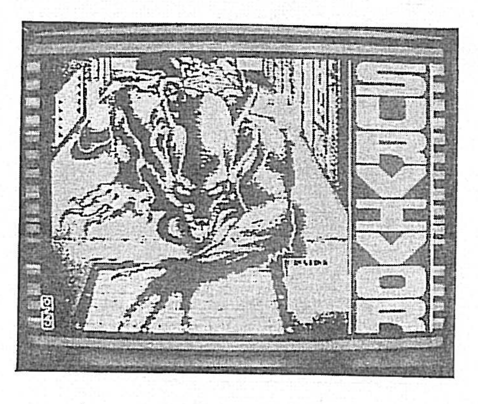


entrar en un ascensor por la puerta o rejilla, es suficiente colocarse frente a ella y agacharse.

La criatura puede aumentar la potencia de sus saltos agachándose para tomar impulso. También puede lanzar el ácido a mayor distancia simplemente con mantener pulsada la tecla de disparo más tiempo.

El aspecto gráfico está muy bien tratado. La pantalla de presentación que nos acompaña durante la carga, es exhibida también al comienzo del juego, acompañada por una pegadiza melodía.

Los movimientos son rápidos y diversos. En lo que respecta al paisaje,



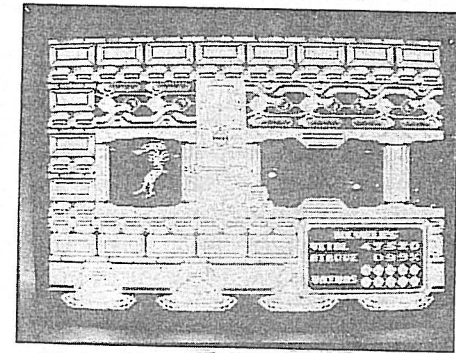
por las ventanas se puede divisar un «triple scroll estelar». Y en lo referente a nuestro personaje, hay que destacar los dos saltos, uno corto a gran altura y otro largo a menor altura, y los desplazamientos a través de los conductos del aire.

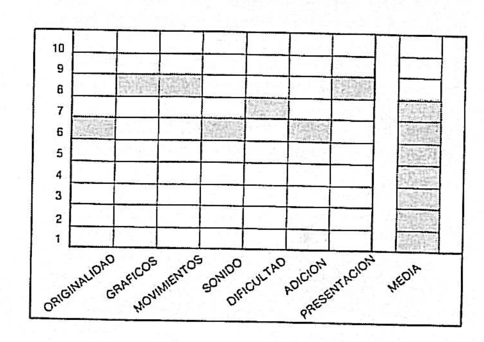
Existe una gran variedad de enemigos, desde seres de raras especies, hasta cañones mecánicos que nos disparan obuses.

Pulsando la tecla RETURN (o ENTER), aparecerá en pantalla una ventana de información que nos indica:

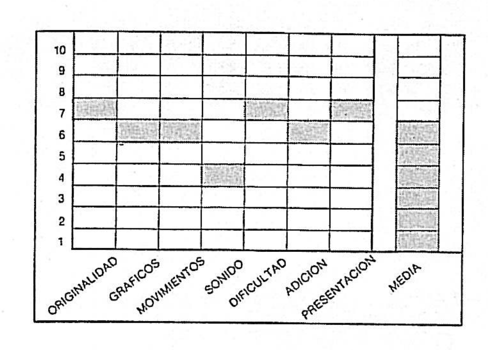
- La energía vital que nos queda.
- Capacidad de ataque en tanto por cien.
- Número de vainas que nos queda por recoger.

En general se trata de un juego atractivo, que suma algo de sonido a la acción y posee un alto grado de dificultad.

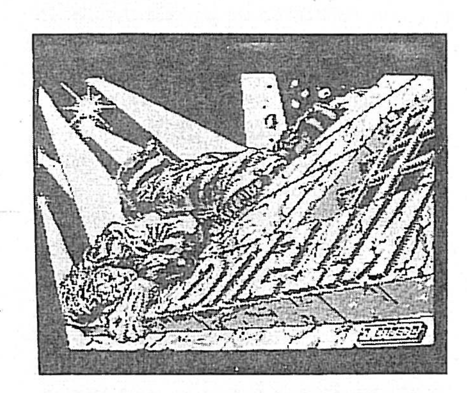




DUSTIN



AVENTURA
DINAMIC



ARGUMENTO

Año 1989.

La cárcel de alta seguridad de WAD-RAS tiene un preso muy especial, se trata de KID SAGUF, más conocido como DUSTIN, el ladrón de joyas y obras de arte más famoso de su tiempo, que finalmente y tras largas persecuciones de la policía ha sido capturado.

Toda la preocupación de sus guardianes está volcada en evitar que DUSTIN pueda escapar.

Por esta razón todo el centro penitenciario ha sido rediseñado para que nada escape al control de su Director; sin embargo, KID SAGUF ha estado durante meses cavilando, analizando horarios, memorizando el funcionamiento interno del centro y ya tiene su plan para conseguir de nuevo las portadas de todos los periódicos.

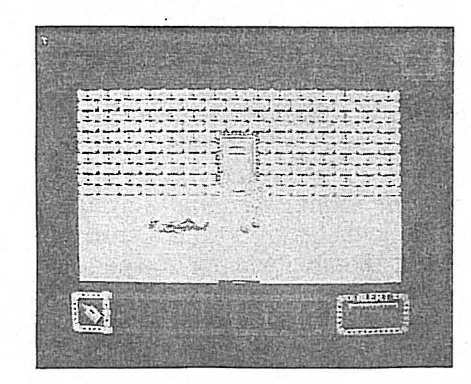
Para no olvidar ningún detalle, ha escrito su desarrollo completo en un cuaderno al que hemos tenido acceso y del que aquí os revelamos su información más interesante.

COMENTARIO

Dustin es un juego que trata un tema bastante original y tiene un alto grado de dificultad, dado el gran número de posibilidades o diferentes situaciones en las que puedes encontrarte, lo cual hace que una vez que has empezado a jugar te sea difícil retirarte de tu ordenador hasta ver si puedes conseguir avanzar un poco más en tu aventura.

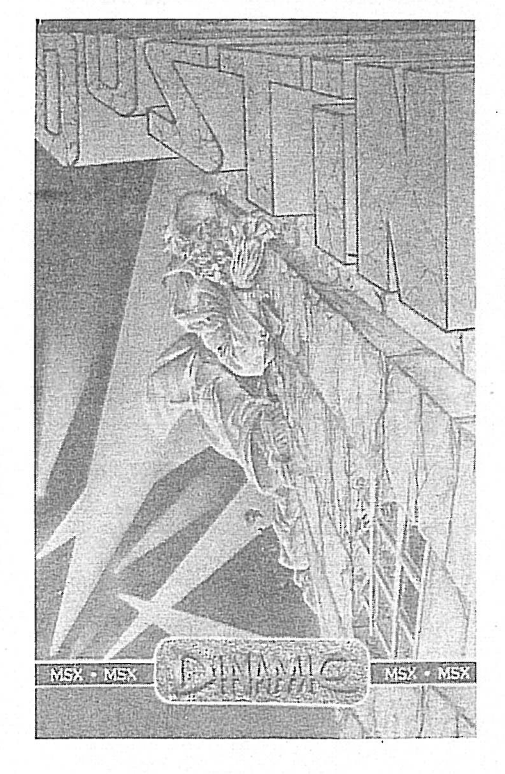
En este juego la portada que acompaña la carga es sumamente atractiva y presenta una perspectiva tridimensional que la hace muy realista.

En las celdas y pasillos que constituyen los gráficos del paisaje, se ha realizado también un efecto tridimensional que les da cierta profundidad. Los personajes se mueven con rapidez y Dustin responde inmediatamente a las instrucciones del teclado. En la parte inferior de la pantalla aparecen, en forma de iconos, los diferentes objetos que vamos obteniendo a lo largo del juego, así como un pequeño marcador en la parte derecha que nos indica el estado de alerta en que nos encontramos.



Los sonidos, si bien existe una bonita melodía durante la presentación, no han sido demasiado trabajados y únicamente nos indican cuando golpeamos o somos golpeados en nuestros enfrentamientos con los guardias de la cárcel.

Dustin es un juego que como ya hemos dicho presenta un alto grado de dificultad, pero esta dificultad está en avanzar a lo largo de las diferentes etapas de la aventura y no en el manejo del protagonista, que es sumamente sencillo.



```
2040 FORA=1TO16:PRINT"çç
          çç";:NEXTA
2050 PRINT"çç  ç       ëëëëëë       çç  çç
çç ççç    ëëë  ëëë    çNOç ççççççççëëëë
ëë  ëëëëëëççëëçççç";
2060 PRINT"ççççççççççççççççççççççççççç
";
2070 VPOKE6192,3
2080 T=500:GOSUB2780
2090 FORA=6TO12:LOCATE2+INT(RND(1)*10),A
:PRINT"ëë":LOCATE15+INT(RND(1)*10),A:PRI
NT"ëë":NEXTA
2100 DEFFNK(Y)=16+Y*8:DEFFNL(X)=8*X-1:DE
FFNG(X,Y)=VPEEK(6146+32*X+Y)
2110 M=1:H=0
2120 X=14:Y=13:X1=40:Y1=40:X2=210:Y2=40:
DX1=4:DY1=4:SPRITEOFF:ONSPRITEGOSUB2580
2130 '
2140 ' ** BUCLE JUEGO **
2150 '
2160 D=STICK(J):IFD=0THENFORA=1TO40:NEXT
A
2170 IFD=1ORD=2ORD=8THENGOSUB2380
2180 IFD=4ORD=5ORD=6THENGOSUB2400
2190 IFD=2ORD=3ORD=4THENGOSUB2440
2200 IFD=8ORD=7ORD=6THENGOSUB2420
2210 SPRITEOFF
2220 IFSTRIG(J)=-1THENGOSUB2490
2230 PUTSPRITE0,(FNK(Y),FNL(X)),1,M-1
2240 X1=X1+DX1:IFX1>208ORX1<40THENDX1=-D
X1
2250 Y1=Y1+DY1:IFY1>131ORY1<40THENDY1=-D
Y1
2260 PUTSPRITE1,(X1,Y1),4,5+M
2270 X2=X2+(2ANDX2<FNK(Y))-(2ANDX2>FNK(Y
)):Y2=Y2+(2ANDY2<FNL(X))-(2ANDY2>FNL(X))
2280 PUTSPRITE2,(X2,Y2),12,7+M
2290 SPRITEON
2300 T=T-1:IFT<0THENT=500:GOSUB2630
2310 LOCATE22,1:PRINTUSING"####";T
2320 SOUND9,10:VPOKE8192,144+(64ANDM=2):
SOUND9,0
2330 M=3-M
2340 GOTO2160
2350 '
2360 ' ** MOVIMIENTO **
2370 '
2380 R=FNG(X-1,Y):GOSUB2530:IFR=0THENR=F
NG(X-1,Y+1):GOSUB2530:IFR=0THENX=X-1
2390 RETURN
2400 R=FNG(X+2,Y):C1=R:GOSUB2530:IFR=0TH
ENR=FNG(X+2,Y+1):GOSUB2530:IFR=0THENX=X+
1
```

— Música al pasar de fase.

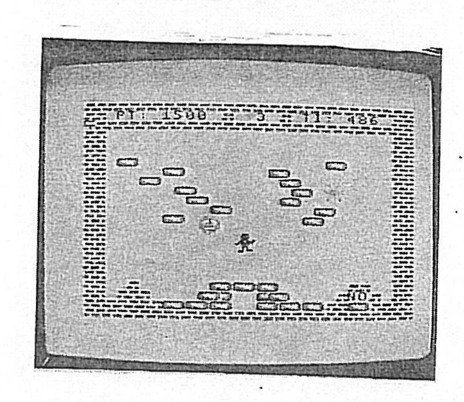
— Sonidos de disparo y explosión.

— E impresión de marcadores.

2800-2900: Rutina de juego finalizado.

2920-2990: Rutina de objetivo conseguido.

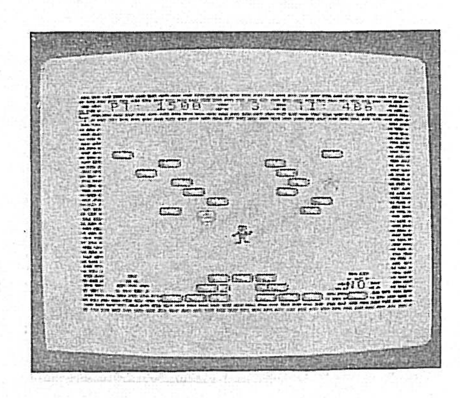
60000-60180: Subrutina de grabación del programa en forma de bloque binario.



Keops

Como nota curiosa a destacar diremos la forma en que se ha resuelto el problema de las colisiones de sprites en la segunda fase. Como existen dos enemigos y mi muñeco puede ocurrir que al comprobar las colisiones por interrupción el programa salte a la rutina de MUERTO al colisionar los dos sprites enemigos.

La solución al problema está clara: retornar al bucle del juego en caso de que la colisión sea entre los dos malos. Pero ¿cómo hacerlo de forma rápida y eficaz?

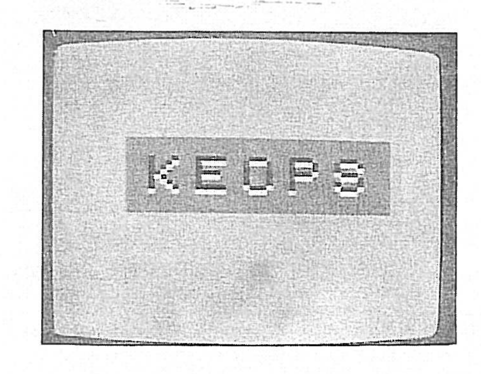


```
2410 RETURN
2420 R=FNG(X,Y-1):GOSUB2530:IFR=0THENR=F
NG(X+1,Y-1):GOSUB2530:IFR=0THENY=Y-1
2430 RETURN
2440 R=FNG(X,Y+2):GOSUB2530:IFR=0THENR=F
NG(X+1,Y+2):GOSUB2530:IFR=0THENY=Y+1
2450 RETURN
2460 '
2470 ' ** COGER O SOLTAR **
2480 '
2490 R=FNG(X+2,Y):R1=FNG(X+2,Y+1)
2500 IFH=0ANDX<14ANDR=136ANDR1=137THENGO
SUB2760:H=1:LOCATE23,20:PRINT"SI":LOCATE
Y,X+2:PRINT"  "
2510 IFH=1ANDR=137ANDR1=136THENGOSUB2760
:BEEP:H=0:LOCATE23,20:PRINT"NO":P=P+50:G
OSUB2780:LOCATEY,X+1:PRINT"@ë":X=X-1:IFX
=13ANDY=13THENRETURN2910
2520 RETURN
2530 IFR<128ORR>137THENR=0
2540 RETURN
2550 '
2560 ' ** COLISION **
2570 '
2580 XX=ABS(X2-X1):IFXX<16THENYY=ABS(Y2-
Y1):IFYY<16THENSPRITEOFF:RETURN
2590 GOSUB2770
2600 '
2610 ' ** MUERTE 2 **
2620 '
2630 SPRITEOFF:GOSUB2770:FORA=10TO12:PUT
SPRITE0,(FNK(Y),FNL(X)),1,A:FORB=1TO250:
NEXTB:NEXTA:FORB=1TO2000:NEXTB:BEEP
2640 V=V-1:GOSUB2780:IFV<1THENRETURN2790
2650 RETURN2120
2660 '
2670 ' ** GRAFICOS **
2680 '
2690 DATA1024,0,127,127,127,0,247,247,24
7,1088,0,127,212,160,208,170,213,127,109
6,0,252,170,6,10,6,74,252
2700 RESTORE2660:FORA=1TO3:READB:FORC=BT
OB+7:READD:VPOKEC,D:NEXTC:NEXTA
2710 VPOKE8208,111:VPOKE8209,23:RETURN
2720 '
2730 ' ** SUBRUTINAS **
2740 '
2750 PLAY"o4cdo3bo4e2r4","o4efdg2r4","o4
gago5c2r4":FORA=0TO3000:NEXTA:BEEP:SOUND
7,56:COLOR1,10,10:CLS:RETURN
2760 SOUND8,12:SOUND1,0:FORA=0TO255STEP1
8:SOUND0,A:NEXTA:SOUND8,0:RETURN
2770 SOUND6,21:SOUND7,247:SOUND8,16:SOUN
```

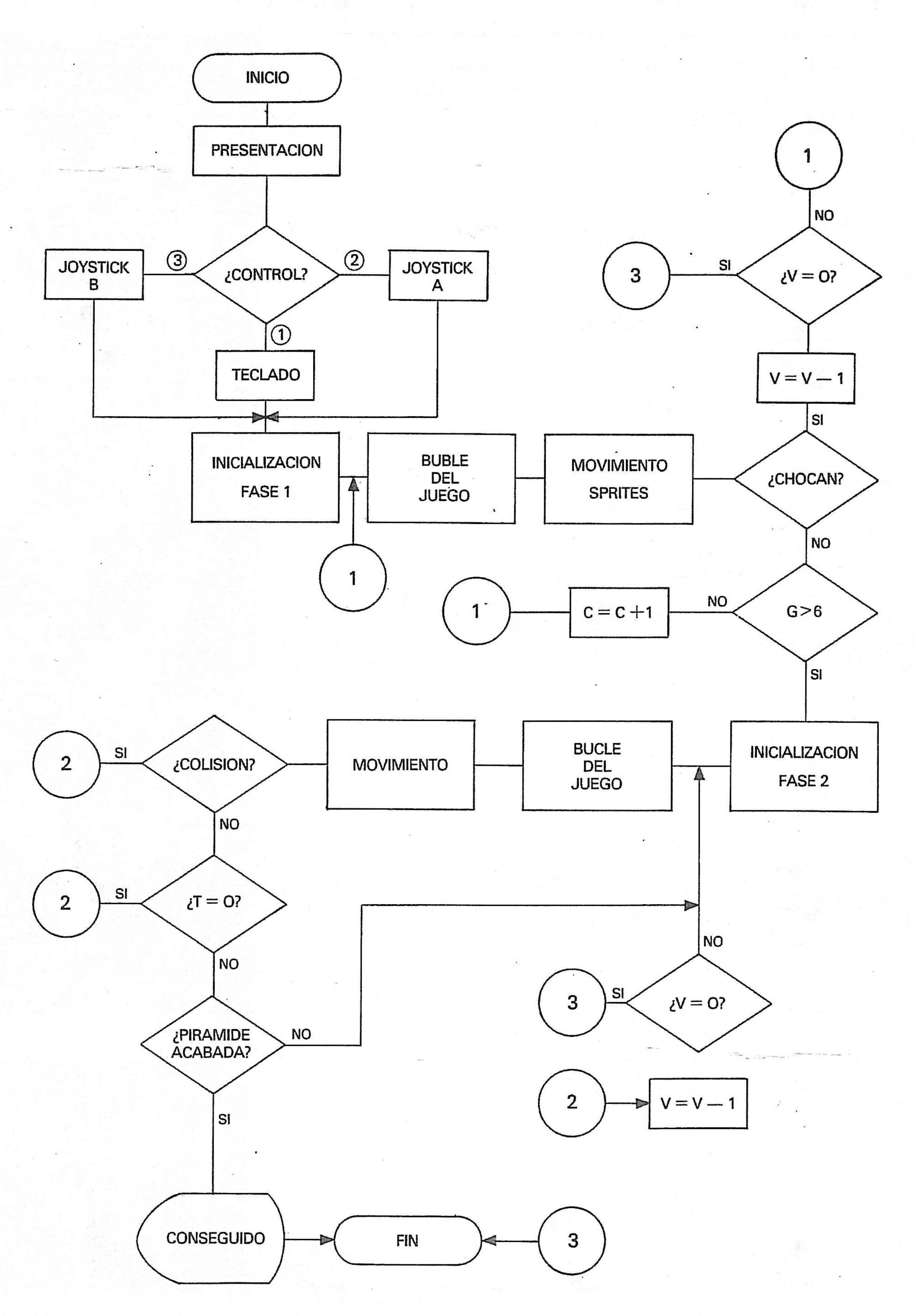
```
D11,100:SOUND12,60:SOUND13,0:RETURN
2780 LOCATE6,1:PRINTUSING"#####";P:LOCAT
E15,1:PRINTUSING"#";V:LOCATE22,1:PRINTUS
ING"####";T:RETURN
2790 '
2800 ' ** GAMEOVER **
2810 '
2820 SCREEN1:COLOR2,1,1:CLS
2830 LOCATE10,5:PRINT"GAME OVER"
2840 LOCATE8,10:PRINT"PUNTOS: ";USING"##
###";P
2850 LOCATE6,16:PRINT"OTRA PARTIDA (S/N)
"
2860 FORA=1TO20:A$=INKEY$:NEXTA
2870 A$=INKEY$
2880 IFA$="N"THENDEFUSR1=0:A=USR1(0)
2890 IFA$="S"THENRUN
2900 GOTO2870
2910 '
2920 ' ** CONSEGUIDO! **
2930 '
2940 M=X:X=FNK(Y):Y=FNL(M)
2950 SOUND8,12:SOUND7,&B111000
2960 FORA=YTO159:PUTSPRITE0,(X,A),1,0:SO
UND0,A:FORB=1TO40:NEXTB:NEXTA:Y=A-1
2970 FORA=1TO15:FORB=1TO255STEPA:SOUND0,
B:NEXTB:PUTSPRITE0,(X,Y),1+RND(1)*9,0:NE
XTA
2980 SOUND8,0
2990 P=P+5000:GOTO2790
3000 '
60000 ' *** GRABACION ***
60005 '
60010 DI=PEEK(&HF6C2)+256*PEEK(&HF6C3)
60020 POKEDI+50,&H26
60030 POKEDI+51,PEEK(&H8002)
60040 POKEDI+52,&H2E
60050 POKEDI+53,PEEK(&H8001)
60060 POKEDI+54,&H22
60070 POKEDI+55,&H1
60080 POKEDI+56,&H80
60090 POKEDI+57,&H21
60100 POKEDI+58,PEEK(&HF6C2)
60110 POKEDI+59,PEEK(&HF6C3)
60120 POKEDI+60,&H22
60130 POKEDI+61,&HC2
60140 POKEDI+62,&HF6
60150 POKEDI+63,&HC3
60160 POKEDI+64,&HAC
60170 POKEDI+65,&H73
60180 PRINT"BSAVE ";CHR$(34);"CAS:KEOPS"
;CHR$(34);",&H8000,";DI+70;CHR$(8);",";D
I+50;
```

La respuesta es tan sencilla como: comprobar que el valor absoluto (ABS) de la diferencia entre las coordenadas X de ambos muñecos, y las coordenadas Y de ellos no exceda nunca de las dimensiones del sprite (16 en nuestro caso). Cuando esta condición se cumpla ambos sprites han colisionado y por lo tanto debemos volver al bucle del juego.

Esperamos que esta solución te sea de utilidad y que por supuesto te diviertas jugando.



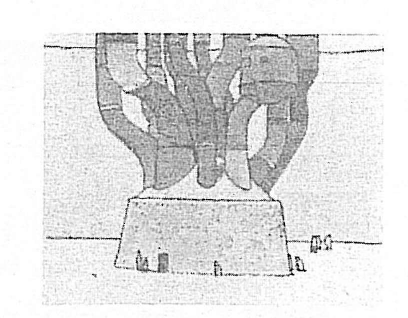
Keops



INICIO
PRESENTACION
JOYSTICK B
③
¿CONTROL?
②
JOYSTICK A
①
TECLADO
INICIALIZACION FASE 1
BUBLE DEL JUEGO
MOVIMIENTO SPRITES
¿CHOCAN?
1
1
NO
3
SI
¿V = O?
V = V — 1
SI
NO
1
C = C + 1
NO
G>6
SI
2
SI
¿COLISION?
MOVIMIENTO
BUCLE DEL JUEGO
INICIALIZACION FASE 2
NO
2
SI
¿T = O?
NO
NO
3
SI
¿V = O?
¿PIRAMIDE ACABADA?
NO
SI
2
V = V — 1
CONSEGUIDO
FIN
3

MGEN (Generador de sprites multicolor):

Este programa está dedicado a todos los usuarios de ordenadores MSX que consideren tediosa la creación de sprites multicolor. Llamamos sprite multicolor al dibujo resultante de superponer varios sprites (monocromos) con sus colores correspondientes.



El programa se encuentra grabado en la cinta en dos partes:

1. Un programa cargador encargado de cargar la segunda parte.
2. Y el programa principal grabado en binario para reducir el tiempo de carga.

```
10 SCREEN 1:KEYOFF:WIDTH29:COLOR2,0,0:LO
CATE5,9:PRINT"CARGANDO: MULTIGEN":BLOAD
"CAS:MGEN",R:END
20 '
30 SAVE "CAS:MGEN"





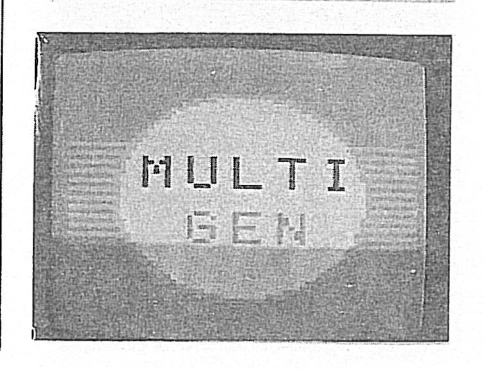

1000 ' GENERADOR DE SPRITES '
1010 ' MULTICOLOR 4 COLORES '
1020 ' ANGEL GARCIA DELGADO '
1030 '
1040 '
1050 '
1060 ' PRESENTACION '
1070 '
1080 CLEAR 2500,57999!:DEFINT A-E,G-I,K-
Z
1090 KEYOFF:SCREEN 3,1,0:COLOR 3,12,12:C
LS:POKE&HFCAB,1
1100 OPEN "GRP:"AS#1
1110 FORA=50TO140STEP8:LINE(0,A)-(256,A)
,2:NEXTA
1120 CIRCLE (127,95),90,8.
1130 PAINT (127,95),9,8
1140 PSET(56,64),9:COLOR15:PRINT#1,"MULT
I"
1150 PSET(56,60),9:COLOR1:PRINT#1,"MULTI
"
1160 PSET(88,114),9:COLOR7:PRINT#1,"GEN"
1170 PSET(88,110),9:COLOR4:PRINT#1,"GEN"
1180 CLOSE
1190 PLAY "t200o4cdefggg.abo5co4aggg.fga
geee.defdccc.","t200o2cdefggg.abo3co2agg
g.fgageee.defdccc.","t200o6cdefggg.abo7c
o6aggg.fgageee.defdccc."
```

Pasos a seguir para grabar el programa en cinta

— Teclear el cargador y pulsar RUN 30 para que se grabe automáticamente en la cinta.

— Teclear ahora el programa principal y hacer RUN 60000 que te imprimirá un BSAVE.

— Sube el cursor hasta la palabra BSAVE y pulsa RETURN. Este programa se cargará a continuación del cargador.



MGEN

El programa se estructura de la siguiente manera:

1000-1210: Se definen las variables enteras y se dibuja la pantalla de presentación.

1220-1360: Se selecciona, por parte del usuario, el tipo de control deseado, teclado o joysticks. Incluye además en la línea 1350 un curioso efecto, que consiste en cambiar la tabla de colores vpokeando valores aleatorios en la VRAM.

```
1200 PLAY "t200o4cdefggg.abo5co4aggg.fga
geee.defdccc.","t200o2cdefggg.abo3co2agg
g.fgageee.defdccc."
1210 FORZ=90TOOSTEP-2:CIRCLE(127,95),Z,1
2:NEXTZ
1220 '
1230 ' FORMA DE CONTROL '
1240 '
1250 SCREEN 1:COLOR 2,1,1
1260 WIDTH 29:CLS:BEEP
1270 LOCATE 0,4:PRINT" -= GENERADOR MULT
ICOLOR =-"
1280 PRINT:PRINT"     1987  UNICORNIO SOF
T":PRINT:PRINT"             A.G.D."
1290 LOCATE 0,13:PRINT"   ELIGE FORMA DE
 CONTROL:"
1300 PRINT:PRINT"           1. TECLADO"
1310 PRINT:PRINT"           2. JOYSTICK A"
1320 PRINT:PRINT"           3. JOYSTICK B"
1330 A$=INKEY$
1340 IFA$>"0"ANDA$<"4"THENJ=VAL(A$)-1:GO
TO 1360
1350 VPOKE8197+6*RND(1),32+16*INT(14*RND
(1)):GOTO 1330
1360 BEEP
1370 '
1380 ' INICIO PROGRAMA '
1390 '
1400 '
1410 ' TIPO DE SPRITE '
1420 '
1430 SCREEN 1,,0:COLOR 1,9,9
1440 WIDTH 29:CLS
1450 LOCATE 0,6,0:PRINT"   ELIGE EL TIPO
 DE SPRITE:":PRINT
1460 PRINT:PRINT"      1. 8x8 SIN AMPLIAR
"
1470 PRINT:PRINT"      2. 8x8 AMPLIADO"
1480 PRINT:PRINT"      3. 16x16 SIN AMPLI
AR"
1490 PRINT:PRINT"      4. 16x16 AMPLIADO"
1500 A$=INKEY$
1510 IFA$>"0"ANDA$<"5"THENT=VAL(A$)-1 EL
SE1500
1520 BEEP
1530 '
1540 ' COLORES A ELEGIR
1550 '
1560 CLS:COLOR 1,3,3
1570 LOCATE 0,10:PRINT"   NUMERO DE COLO
RES (1-4)"
1580 A$=INKEY$:IFA$>"0"ANDA$<"5"THENNC=V
AL(A$) ELSE1580
```

```
1590 BEEP
1600 CLS:SCREEN 1,T:COLOR 1,10,10
1610 '
1620 ' DEFINICION DE UDGS '
1630 '  E INICIALIZACION  '
1640 '
1650 FORZ=1024TO1280STEP64:VPOKEZ,255:FO
RN=Z+1TOZ+6:VPOKEN,129:NEXTN:VPOKEN,255:
NEXTZ
1660 FORZ=1032TO1039:VPOKEZ,0:NEXTZ
1670 VPOKE8208,31:VPOKE8209,20:VPOKE8210
,25:VPOKE8211,19:VPOKE8212,23
1680 IFT<2THENN=256/NC:M=8 ELSEN=64/NC:M
=16
1690 DIM D$(N),C(M,M)
1700 T1=4:T2=8:T3=2:T4=7
1710 FORX=1TOM:FORY=1TOM
1720 LOCATEY,X,0:PRINTCHR$(128):C(X,Y)=0
1730 NEXTY:A$=INKEY$:NEXTX
1740 LOCATE 18,1:PRINT"COLOR: 0 ";CHR$(1
28)
1750 LOCATE 19,3:PRINT"MUESTRA:"
1760 LOCATE 22,5:PRINT"NO"
1770 FORZ=7TO12:LOCATE20,Z:PRINT"üüüüüü"
:NEXTZ
1780 GOSUB3290
1790 S=0:C=0:X=1:Y=1
1800 '
1810 ' BUCLE PRINCIPAL '
1820 '
1830 D=STICK(J)
1840 IFD=8ORD=1ORD=2THENX=X-1:IFX<1THENX
=1
1850 IFD=6ORD=5ORD=4THENX=X+1:IFX>MTHENX
=M
1860 IFD=2ORD=3ORD=4THENY=Y+1:IFY>MTHENY
=M
1870 IFD=8ORD=7ORD=6THENY=Y-1:IFY<1THENY
=1
1880 LOCATE Y,X,1
1890 POKE&HF3F7,1:POKE&HF3F6,1:T$=INKEY$
1900 IFT$="0"THENE=0
1910 IFT$="1"THENE=1
1920 IFNC=1THEN 1980
1930 IFT$="2"THENE=2
1940 IFNC=2THEN 1980
1950 IFT$="3"THENE=3
1960 IFNC=3THEN 1980
1970 IFT$="4"THENE=4
1980 IFSTRIG(J)=-1THENC(X,Y)=E
1990 IFT$="T"THENGOSUB2100
2000 IFT$="I"THENGOSUB2230
2010 IFT$="C"THENBEEP:GOTO1710
```

1370-1790: Es donde comienza realmente el programa. En este bloque de líneas es donde se selecciona el tipo de sprite a generar, el número de colores del sprite y se definen los gráficos y se inicializan las tablas para almacenar los datos resultado del programa.

El número de colores a elegir está comprendido entre 1 y 4 por ser 4 el número máximo de sprites que se pueden poner en línea. Por supuesto, el programa generará para cada sprite multicolor un número de sprites mo-

MGEN

nocromo igual al número de colores elegidos que se almacenan contiguos.

1800-2090: En estas líneas es donde se encuentra el bucle principal del programa, que inspecciona el teclado y accede a las subrutinas seleccionadas por nosotros. Se encarga también de imprimir el cursor de la celdilla de gráficos.

2100-2220: Este bloqueo como todos los que vienen a continuación son subrutinas llamadas desde el bucle principal. Esta

```
2020 IFT$="V"THENGOSUB2590
2030 IFT$="E"THENGOSUB2700
2040 IFT$="W"THENGOSUB2860
2050 IFT$="R"THENLOCATE 0,0,0:RUN 1220
2060 IFT$="S"THENGOSUB2940
2070 LOCATE 24,1,0:PRINTE;CHR$(128+E*8);
2080 LOCATEY,X,0:PRINTCHR$(128+C(X,Y)*8)
;
2090 GOTO 1830
2100 '
2110 ' CAMBIAR COLORES '
2120 '
2130 GOSUB 3240
2140 INPUT "COLOR DEL FONDO (0):";A:VPOK
E8208,16+A
2150 GOSUB 3240:INPUT "COLOR TINTA 1:";T
1:VPOKE8209,16+T1
2160 IFNC=1THEN2220
2170 GOSUB 3240:INPUT "COLOR TINTA 2:";T
2:VPOKE8210,16+T2
2180 IFNC=2THEN2220
2190 GOSUB 3240:INPUT "COLOR TINTA 3:";T
3:VPOKE8211,16+T3
2200 IFNC=3THEN2220
2210 GOSUB 3240:INPUT "COLOR TINTA 4:";T
4:VPOKE8212,16+T4
2220 GOSUB3290:E=0:PLAY "C","D","E":RETU
RN
2230 '
2240 ' INTRODUCIR SPRITE '
2250 '
2260 GOSUB 3240
2270 INPUT "EN QUE SPRITE:";NS
2280 IFNS<0ORNS>NTHENPLAY"C":GOTO2260
2290 A$=""
2300 FORB1=1TONC
2310 PRINT" SPRITE: ";B1
2320 IFM=8THENGOSUB2370
2330 IFM=16THENGOSUB2400
2340 NEXTB1
2350 D$(NS)=A$:S=NS:GOSUB2470
2360 GOSUB3290:RETURN
2370 FORB2=1TO8:V=0:FORB3=1TO8:IFC(B2,B3
)=B1THENV=V+2^(8-B3)
2380 NEXTB3:A$=A$+CHR$(V):NEXTB2
2390 RETURN
2400 FORB4=1TO9STEP8:FORB5=1TO9STEP8:FOR
B2=B5TOB5+7:V=0:W=7:FORB3=B4TOB4+7
2410 IFC(B2,B3)=B1THENV=V+(2^W)
2420 W=W-1
2430 NEXT B3:A$=A$+CHR$(V):NEXT B2
2440 NEXT B5
2450 NEXT B4
```

```
2460 RETURN
2470 '
2480 ' MUESTRA =S '
2490 '
2500 LOCATE 21,5:PRINTS;"  ";
2510 MM=M+(16 AND M=16):FORZ=1TONC
2520 A$=MID$(D$(S),1+MM*(Z-1),MM):SPRITE
$(Z-1)=A$
2530 NEXTZ:A$="":FORZ=1TOM:A$=A$+CHR$(0)
:NEXTZ
2540 PUTSPRITE0,(184,63),T1,0
2550 IFNC>1THENPUTSPRITE1,(184,63),T2,1
ELSE SPRITE$(1)=A$
2560 IFNC>2THENPUTSPRITE2,(184,63),T3,2
ELSE SPRITE$(2)=A$
2570 IFNC>3THENPUTSPRITE3,(184,63),T4,3
ELSE SPRITE$(3)=A$
2580 RETURN
2590 '
2600 ' VISUALIZACION '
2610 '
2620 GOSUB3240:PRINT" CURSOR ARRIBA. SIG
UIENTE":PRINT
2630 PRINT"  CURSOR ABAJO. ANTERIOR":PRI
NT:PRINT"  SPACE. FINALIZAR"
2640 I=STICK(J)
2650 IFI=1THENS=S+1:IFS>NTHENS=1
2660 IFI=5THENS=S-1:IFS<0THENS=N
2670 GOSUB2470
2680 IFSTRIG(J)=-1THENBEEP:GOSUB 3290:RE
TURN
2690 GOTO 2640
2700 '
2710 ' ESPEJO '
2720 '
2730 BEEP
2740 FORB1=1TOM:FORB2=1TOM/2
2750 SWAPC(B1,B2),C(B1,M+1-B2)
2760 NEXTB2:NEXTB1
2770 GOSUB 2790:FORZ=1TO15:I$=INKEY$:NEX
TZ
2780 RETURN
2790 '
2800 ' IMPRIME CELDILLA '
2810 '
2820 BEEP
2830 FORB1=1TOM:FORB2=1TOM
2840 LOCATEB2,B1,0:PRINTCHR$(128+8*C(B1,
B2))
2850 NEXTB2:NEXTB1:RETURN
2860 '
2870 ' INVERTIR '
2880 '
```

concretamente cumple la misión de cambiar los colores de las tintas que usamos. Recuerda que para borrar has de pintar con tinta 0.

2230-2460: Subrutina que guarda el sprite generado en una tabla.

2470-2580: Imprime las muestras.

2590-2690: Subrutina para poder visualizar la muestra deseada.

MGEN

2700-2780: Subrutina de espejo horizontal.

2790-2850: Imprime la celdilla con su contenido.

2860-2930: Subrutina de espejo vertical (invertir).

2940-3230: Rutina de grabación que además de los datos de lo sprites graba también una subrutina en c/m que pasa estos datos instantáneamente a la VRAM. La rutina comienza en la dirección 58000 y finaliza en la

```
2890 BEEP
2900 FORB1=1TOM:FORB2=1TOM/2
2910 SWAPC(B2,B1),C((M+1)-B2,B1)
2920 NEXTB2:NEXTB1
2930 GOTO 2770
2940 '
2950 ' GRABACION (BSAVE) '
2960 ' INCLUYENDO RUTINA '
2970 '  EN C/M  QUE LOS  '
2980 ' GUARDA EN LA VRAM '
2990 '
3000 '  COMIENZO: 58000  '
3010 '  FINAL    : 60062  '
3020 '
3030 ' LLAMADA A RUTINA: '
3040 '         58000      '
3050 '
3060 GOSUB3240:INPUT"SPRITE INICIAL:";SI
3070 GOSUB3240:INPUT"SPRITE FINAL:";SF
3080 DATA 1,0,8,17,0,56,33,157,226,205,9
2,0,201
3090 RESTORE 3080:FORF=58000!TO58012!:RE
ADG:POKEF,G:NEXTF
3100 F=58013!:A$="":FORG=1TOM+(16ANDM=16
):A$=A$+CHR$(0):NEXTG
3110 FORG=SITOSF
3120 I$=D$(G):I=LEN(I$):IFI=0THEND$(G)=A
$:GOTO3120
3130 FORH=1TOI:POKEF,ASC(MID$(I$,H,1)):F
=F+1:NEXTH
3140 NEXTG
3150 GOSUB3240:PRINT"LO GRABAS? (S/N):"
3160 A$=INKEY$
3170 IFA$="N"THENGOSUB3290:RETURN
3180 IFA$<>"S"THEN3160
3190 GOSUB3240:INPUT"NOMBRE:";N$
3200 PRINT:PRINT" PULSA RECORD Y SPACE"
3210 IFINKEY$<>" "THEN3210
3220 BSAVE N$,58000!,60062!
3230 GOSUB3290:RETURN
3240 '
3250 ' BORRADO VENTANA '
3260 '      DE TEXTO      '
3270 '
3280 FORA=18TO23:LOCATE 0,A,0:PRINT"
                                   ";:NEXTA:LOCATE
1,18,0:FORA=1TO10:Z$=INKEY$:NEXTA:RETURN
3290 '
3300 ' INSTRUCCIONES '
3310 '
3320 GOSUB3240
3330 PRINT"  0-4. SELECIONAN TINTAS"
3340 PRINT"    C. CLS  SPACE. PINTA"
```

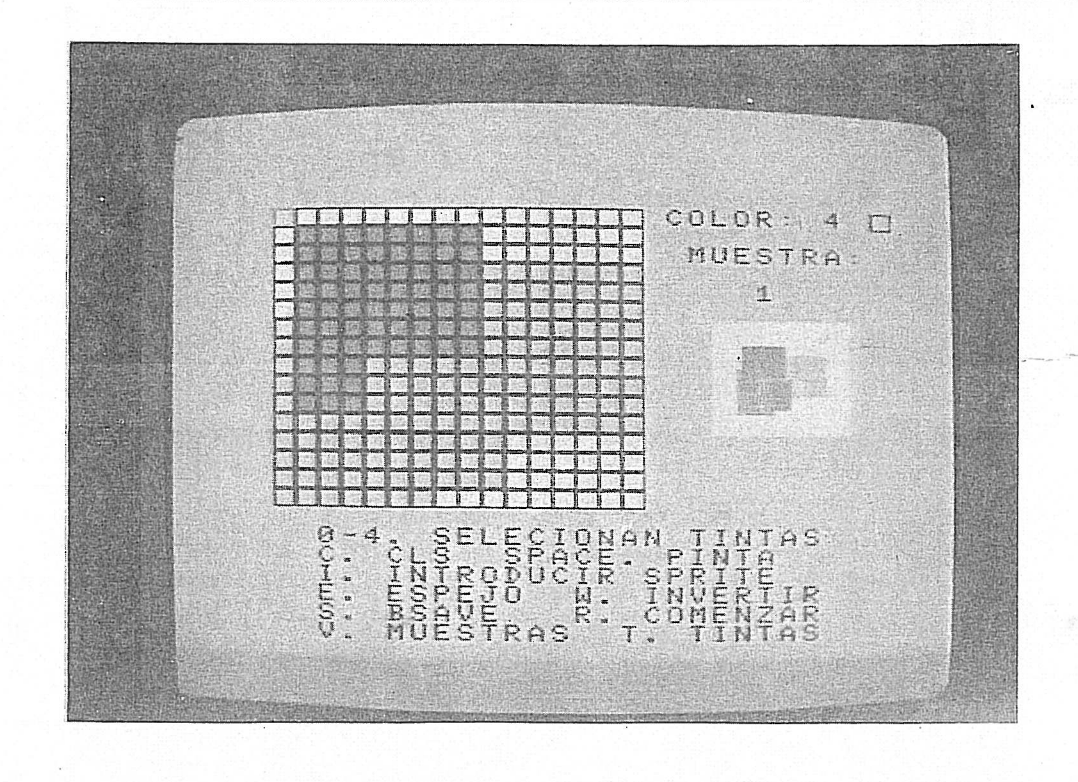
```
3350 PRINT"    I. INTRODUCIR SPRITE"
3360 PRINT"    E. ESPEJO  W. INVERTIR":PR
INT"    S. BSAVE   R. COMENZAR"
3370 PRINT"    V. MUESTRAS  T. TINTAS";
3380 RETURN
3390 '
3400 '
3410 '
3420 '
60000 ' *** GRABACION ***
60010 DI=PEEK(&HF6C2)+256*PEEK(&HF6C3)
60020 POKEDI+50,&H26
60030 POKEDI+51,PEEK(&H8002)
60040 POKEDI+52,&H2E
60050 POKEDI+53,PEEK(&H8001)
60060 POKEDI+54,&H22
60070 POKEDI+55,&H1
60080 POKEDI+56,&H80
60090 POKEDI+57,&H21
60100 POKEDI+58,PEEK(&HF6C2)
60110 POKEDI+59,PEEK(&HF6C3)
60120 POKEDI+60,&H22
60130 POKEDI+61,&HC2
60140 POKEDI+62,&HF6
60150 POKEDI+63,&HC3
60160 POKEDI+64,&HAC
60170 POKEDI+65,&H73
60180 PRINT"BSAVE";CHR$(34);"CAS:MGEN";C
HR$(34);",&H8000,";DI+70;CHR$(8);",";DI+
50
```

58012. Los datos de los sprites se almacenan a partir de la dirección 58013 y pueden llegar hasta la 60062.

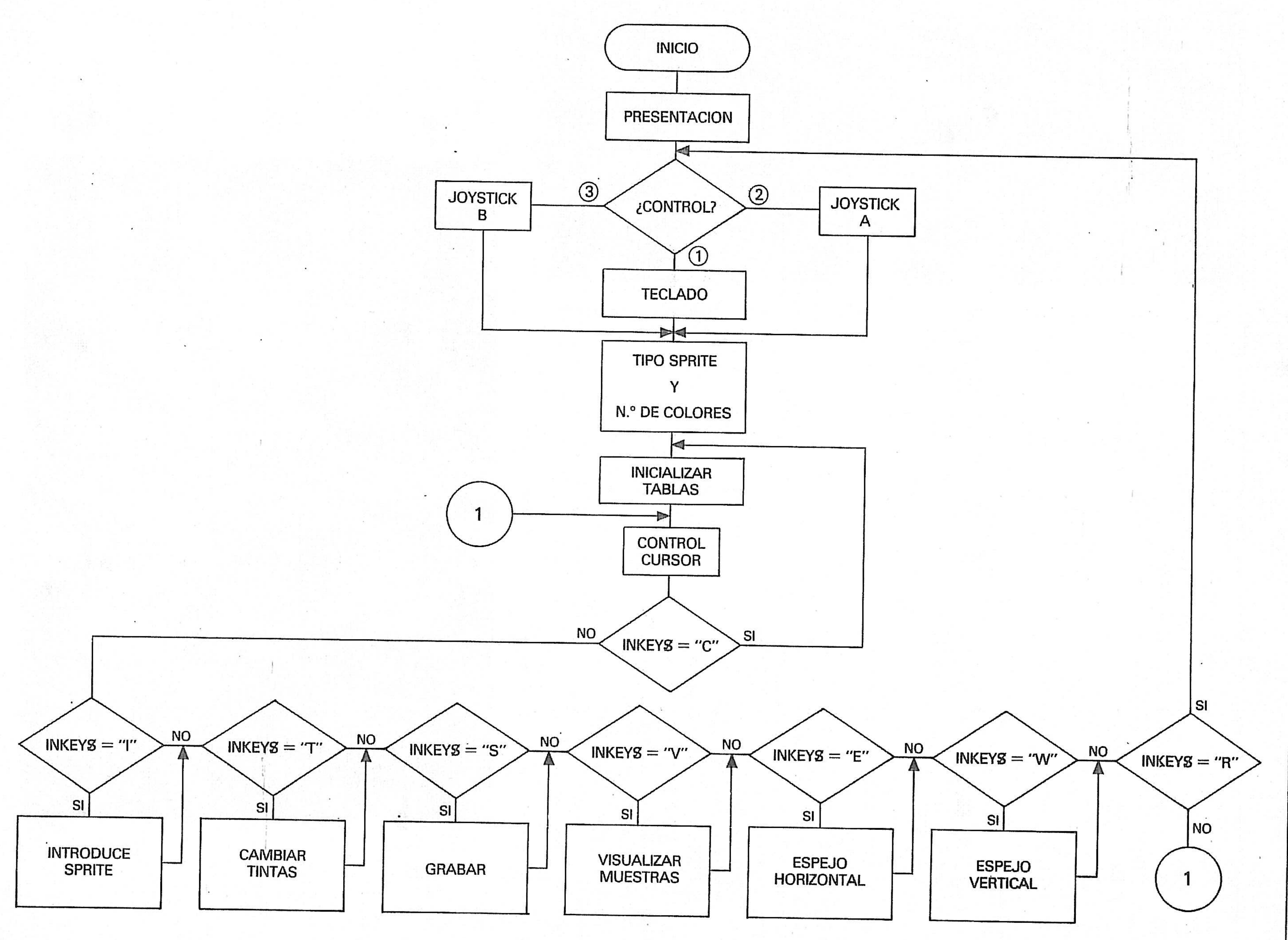
3240-3420: Subrutinas que borran e imprimen las instrucciones.

60000-60180: Rutina que graba el programa con BSAVE.

NOTA: El número de sprites que se pueden generar depende tanto del tamaño como del número de colores.

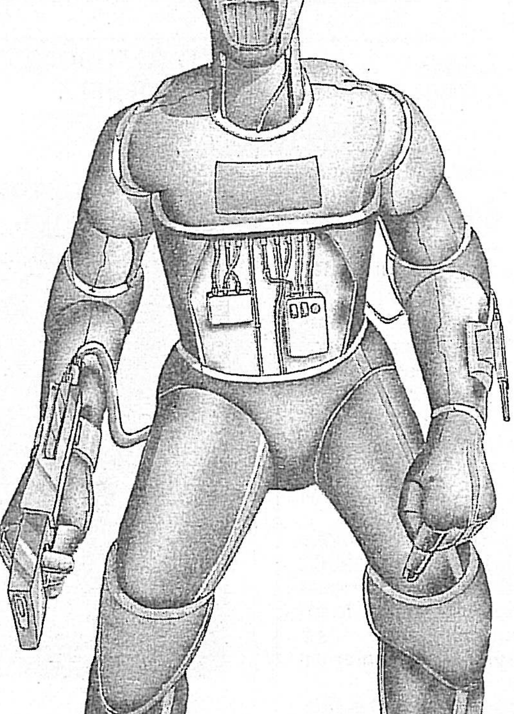


MGEN



INICIO
PRESENTACION
¿CONTROL?
①
②
③
JOYSTICK A
TECLADO
JOYSTICK B
TIPO SPRITE Y N.° DE COLORES
INICIALIZAR TABLAS
CONTROL CURSOR
1
INKEY$ = "C"
SI
NO
INKEY$ = "I"
SI
INTRODUCE SPRITE
NO
INKEY$ = "T"
SI
CAMBIAR TINTAS
NO
INKEY$ = "S"
SI
GRABAR
NO
INKEY$ = "V"
SI
VISUALIZAR MUESTRAS
NO
INKEY$ = "E"
SI
ESPEJO HORIZONTAL
NO
INKEY$ = "W"
SI
ESPEJO VERTICAL
NO
INKEY$ = "R"
SI
NO
1

PROXIMAS SECCIONES



CORREO DEL LECTOR

Consultas, dudas, problemas sugerencias, etc. de los lectores que esten relacionados con el sistema MSX, serán publicadas y respondidas en esta sección.

CLUB

En este apartado daremos cabida a todos los anuncios entre particulares o clubs de informática que esten interesados en cambiar, comprar o vender algo, e establecer contactos, etc.

Enviad vuestros mensajes: Mundo MSX, Tomás López, 3 6.° 28009 MADRID



MSX. GUIA DEL USUARIO



Autor: Paul Hoffman
Editorial: McGraw-Hill

Este libro contiene información sobre cómo utilizar el ordenador MSX, cómo programar en MSX-BASIC y, si se dispone de unidad de disco, cómo emplear el sistema MSX-DOS.

Aun cuando el libro está concebido de modo que los usuarios principales puedan comprenderlo con facilidad, contiene también gran cantidad de información útil para usuarios expertos.

Destacan los siguientes temas:

— Introducción a las computadoras para los principiantes: Contiene descripciones no técnicas de lo que es una computadora y de cómo comenzar a utilizarla.

— Visión general del mercado de MSX: Programas, equipos y accesorios.

— Lista completa de todas las órdenes del MSX-BASIC: Examinadas en grupos funcionales para facilitar la relación existente entre ellas. Se proporcionan programas ejemplos.

— Introducción en profundidad al MSX-DOS: Amplia información sobre todos los conceptos que sirven de base al MSX-DOS.

— Tablas de referencias rápidas: Todas las órdenes del MSX-BASIC y del MSX-DOS. Así como una tabla de caracteres y un examen de los aspectos técnicos del sistema.

30 RUTINAS DE UTILIDAD EN CODIGO MAQUINA



Autor: Steve Webb
Editorial: Rama
Páginas: 96

Este elemental y práctico librito nos introduce en el mundo del Código Máquina de una manera amena, eficaz y sin necesidad de tener ningún conocimiento previo sobre el tema.

Las rutinas y sugerencias que contiene serán muy apreciadas por aquellos usuarios que, habiéndoseles quedado pequeño el BASIC, quieran dar a sus programas, sobre todo de juegos, mayor rapidez y vistosidad, ocupándoles menos memoria.

Los primeros capítulos tratan temas como memoria, ROM, SPRITES y memoria de vídeo.

En los siguientes capítulos se describen las rutinas que se pueden utilizar para la realización de programas en Basic. Por ejemplo:

— Mover la pantalla en cuatro direcciones.

— Mover por separado una fila o columna de caracteres.

— Trasladar caracteres de forma que se pueda escribir texto a los lados o incluso la parte inferior de la pantalla.

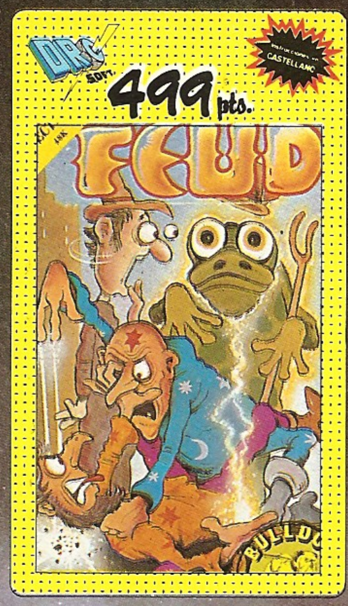
— Añadir efectos de sonido.

Por último, la mayoría de las rutinas pueden ser alteradas para adaptarlas a necesidades personales.

SERVICIO DE INFORMACION AL LECTOR

PONEMOS A DISPOSICION DE NUESTROS LECTORES UN NUEVO SERVICIO GRATUITO DE INFORMACION SOBRE CUALQUIER PRODUCTO QUE DIRECTA O INDIRECTAMENTE SEA TRATADO EN CUALQUIERA DE LAS PAGINAS DE MUNDO MSX

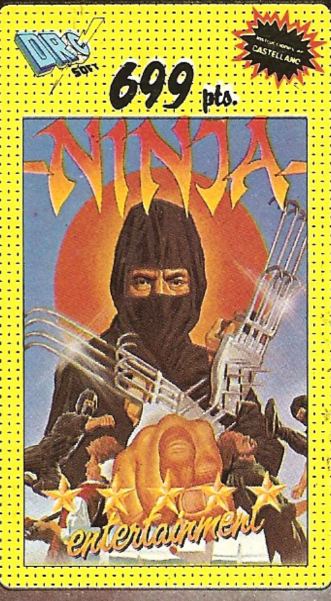
Información que solicito: ....
....
Mencionado en revista n.º: .... Página n.º: ....
....
Nombre y Apellidos: ....
Dirección: ....
Localidad: .... C.P. .... Provincia: ....
Modelo ordenador: .... Fecha ....





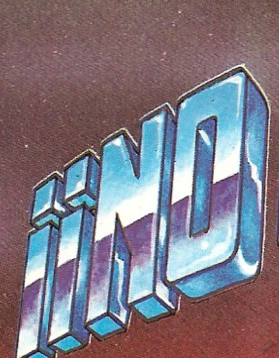
COMMODORE
SPECTRUM
AMSTRAD
MSX

FEUD
Con tu sabiduría puedes encontrar los ingredientes de las POCIONES MAGICAS, mientras recitas los conjuros de tu maligno enemigo LEANORIC.



COMMODORE
SPECTRUM
AMSTRAD

NINJA
Eres un NINJA, demuestra tu destreza luchando contra los mejores Budokas en un combate a MUERTE.

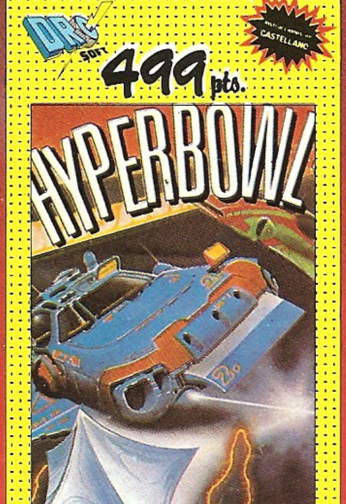


¡¡NO PUEDE SER!!

499 pts.

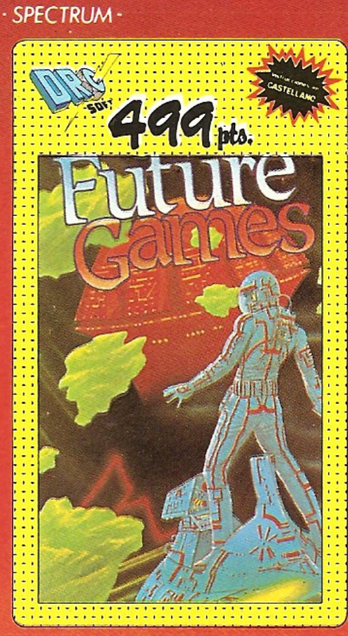
SERIE M.A.D. 699 pts.

COMMODORE
SPECTRUM
AMSTRAD
MSX

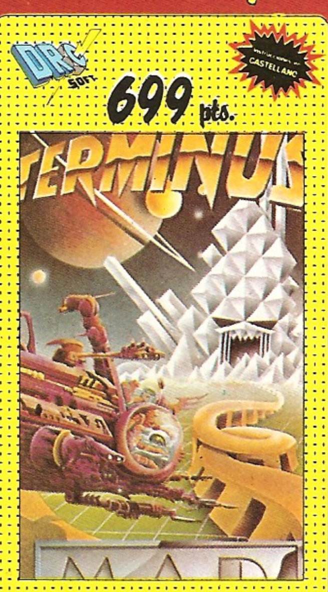


HYPERBOWL
En el año 3.600 el hockey sobre hielo ha alcanzado el nivel de máxima tecnología. Diez clases diferentes de naves, se enfrentan en un campo metálico...

- SPECTRUM -



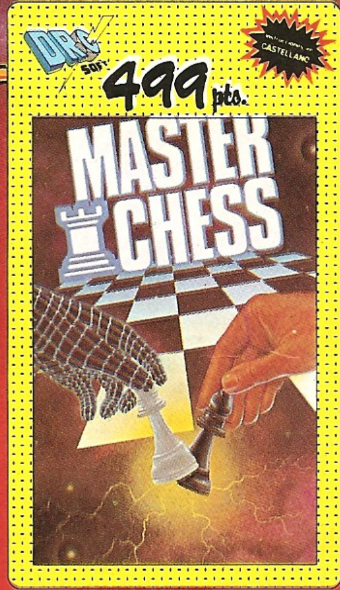
FUTURE GAMES
Juega tu libertad contra la muerte. Si ganas, la consigues, si pierdes, desde el lugar donde estés, ya no te importará.



SPECTRUM
AMSTRAD
MSX

TERMINUS
Un grupo de gamberros, que siembra el pánico por donde pasa, ha decidido rescatar a su lider del planeta prisión TERMINUS

COMMODORE



MASTER CHESS
Tu ordenador es tu mejor contrincante. Piensa más rápido que tu, pero tu puedes jugar mejor y ganarle. ¡INTENTALO!

Y otros 120 juegos mas.

MASTERTRONIC




SPECTRUM • AMSTRAD CPC • COMMODORE • MSX
FREDDY HARDEST
FICHA TECNICA
• Dominio de las artes marciales: Patada - Puñetazo - Protección.
• Turbo-Laser de carga atmosférica.
• Salto controlable en altura y dirección.
• 30 Pantallas de Multiscroll.
• 64 Pantallas conectadas por ascensores y túneles.
• 16 Terminales de ordenador para hacer de Hacker.
• Enemigos inteligentes que se protegen de tu Laser.
• 5 Espécies de animales hostiles.
• 3 Tipos de Androides asesinos.
FX DOBLE CARGA
Signfica dos programas en uno, el doble de acción, el doble de aventura, el doble de gráficos y máxima adictividad.Prueba un FX DOBLE CARGA de DINAMIC, te sorprenderá.
SPECTRUM
COMMODORE
AMSTRAD CPC
MSX
FREDDY HARDEST, agente secreto y refutado playboy, se encuentra en un satélite enemigo con su nave averiada. Para escapar, deberá llegar hasta la base enemiga y robar un cazo. FREDDY, demostrará su destreza saltando, agarrándose a las argollas que encuentre, trepando por cuerdas, disparando su Laser,... FREDDY, es todo un número uno, pero sobre todo, quiere escapar vivo.
875
DINAMIC
DINAMIC SOFTWARE. PZA. DE ESPAÑA, 18. TORRE DE MADRID, 29-1. 28008 MADRID. TELEX: 44124 DSOFT-E
TIENDAS Y DISTRIBUIDORES: (91) 314 • 18 • 04. PEDIDOS CONTRA REEMBOLSO: (91) 248 • 78 • 87.